Simultaneous Reconstruction and Segmentation with Class-Specific Priors

Romanov, Mikhail

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Simultaneous Reconstruction and Segmentation with Class-Specific Priors

Mikhail Romanov

Kongens Lyngby 2015

Technical University of Denmark
Department of Applied Mathematics and Computer Science
Richard Petersens Plads, building 324,
2800 Kongens Lyngby, Denmark
Phone +45 4525 3031
compute@compute.dtu.dk
www.compute.dtu.dk

Summary (English)

Studying the interior of objects using tomography often require an image segmentation, such that different material properties can be quantified. This can for example be volume or surface area. Segmentation is typically done as an image analysis step after the image has been reconstructed. This thesis investigates computing the reconstruction and segmentation simultaneously. The advantage of this is that because the reconstruction and segmentation are computed jointly, reconstruction errors are not propagated to the segmentation step. Furthermore the segmentation procedure can be used for regularizing the reconstruction process. The thesis provides models and algorithms for simultaneous reconstruction and segmentation and their performance is empirically validated.

Two method of simultaneous reconstruction and segmentation are described in the thesis. Also, a method for parameter selection is given. The reconstruction and segmentation are modeled as two parts: the image that is reconstructed and a so-called Hidden Markov Measure Field Model (HMMFM). Pixel values in the image contain material attenuation coefficients and the HMMFM contains pixelwise probabilities for material classes. The number of material classes and their parameters are assumed known a priori. These parameters are the mean value of the class attenuation coefficients and their standard deviations. Given this input together with projection data, the problem is to find the image and HMMFM. The segmentation is obtained from the HMMFM as the most probable class in each pixel.

The solution for the reconstruction and segmentation problem is found using an

algorithm that simultaneously minimizes the reprojection error, deviation of the grey levels of pixels from known mean values and the spatial differences in the class probabilities.

In the first Simultaneous Reconstruction and Segmentation (SRS) method data is assumed Gaussian distributed and the minimization is done using standard optimization techniques in two stages. Experimental validation on both phantom and real data shows that modeling the reconstruction and segmentation simultaneously has superior performance, especially when the problem is underdetermined, i.e. when the number of unknowns in the reconstruction exceeds the number of observations.

The second SRS method assumes Poisson distributed data, which is the case for data originating from discrete events like photon counts. The algorithm is again based on solving a minimization problem. In addition a relaxation strategy is employed in order to avoid being stuck in local minimum. This model is also validated on artificial data.

Selecting appropriate regularization parameters can be difficult, so the last thing that we consider is a parameter selection approach. The most promising approach was a modified L-curve algorithm, which was empirically analyzed.

This thesis contributes with methods for simultaneous reconstruction and segmentation and demonstrates the benefits of this approach in situations where only few projections are available and data is noisy. Here a higher precision image as well as segmentation can be computed.

Summary (Danish)

Studier af objekters indre struktur ved brug af tomografi kræver ofte segmentering, så forskellige materialeegenskaber kan blive kvantificeret. Dette kan eksempelvis være volumen eller overfladeareal og lignende. Segmenteringen gøres typisk ved et billedanalyseskridt, efter billedet er blevet rekonstrueret. I denne afhandling undersøges teknikker til at rekonstruere og segmentere simultant. Fordelen er at kun et skridt er nødvendigt, og herved undgås at rekonstruktionsfejl propagerer til segmenteringen. I tillæg hertil kan segmenteringsproceduren bruges til at regularisere rekonstruktionsprocessen. Afhandlingen giver modeller og algoritmer til simultan rekonstruktion og segmentering, og disse er empirisk valideret.

To metoder til simultan rekonstruktion og segmentering (SRS) er beskrevet i denne afhandling samt en metode til parameterudvælgelse. Rekonstruktionen og segmenteringen er modelleret i to dele, nemlig billedet som rekonstrueres samt en såkaldt Hidden Markov Measure Field Model (HMMFM). Pixelværdier i billedet angiver dæmpningskoefficienter, og HMMFM'en indeholder pixelvise sandsynligheder for materialeklasser. Antallet af materialeklasser og deres parametre er antaget kendt a priori. Disse parametre er middelværdier og standardafvigelser. Med dette input samt det målte projektionsdata, hvilket er radiografier af et objekt, er problemet at finde billedet og HMMFM-parametrene. Segmenteringen er givet fra HMMFM som den mest sandsynlige klasse.

Rekonstruktions- og segmenteringsproblemet er løst ved hjælp af en algoritme, som på samme tid minimerer reprojektionsfejlen, afvigelse i gråtoneværdier fra kendte middelværdier og den rumlige forskel i sandsynligheder for klasserne.

I den første SRS-metode antages data gaussisk fordelt og minimeringen gennemføres med standard optimeringsmetoder i to stadier. Eksperimentel validering på både fantomdata og virkeligt data viser, at modellering af både rekonstruktionen og segmenteringen på samme tid forbedrer performance, især når problemet er underbestemt, det vil sige at antallet af ukendte i rekonstruktionen overstiger antallet af observationer.

I den anden SRS-metode antages data at være Poisson-fordelt, hvilket typisk gælder for data, som kommer fra en diskret proces som eksempelvis fotontælling. Algoritmen er igen baseret på at løse et minimeringsproblem. I tillæg hertil benyttes en relakseringsstrategi til at undgå lokale minima. Denne model er valideret på fantom-data.

Valget af passende regulariseringsparametre kan være vanskeligt, så den sidste metode omhandler parametervalg. Den mest lovende tilgang var en modificeret L-kurve-algoritme, som er blevet analyseret empirisk.

Bidraget i denne afhandling er metoder til simultan rekonstruktion og segmentering, og fordelene ved denne tilgang er demonstreret i situationer, hvor kun få projektioner er tilgængelige, og data er støjfuldt. I disse tilfælde bliver både det rekonstruerede billede og rekonstruktionen mere præcis.

Preface

This thesis was prepared at DTU Compute in fulfilment of the requirements for acquiring an PhD in Scientific Computing.

The thesis deals with problem of simultaneous reconstruction and segmentation using class priors known in advance.

The thesis consists of nine chapters. In the first chapter we overview the reasons why the tomography is needed, then we talk about the physical background of tomography and measurement techniques. In chapter 3 we discuss classical reconstruciton techniques. Then we move to the classical segmentation techniques. After that follows the overview of related work that joins reconstruction with segmentation. In chapter 6 we introduce Simultaneous Reconstruction and Segmentation algorithm with class priors. After that we introduce the relaxed SRS modification algorithm. Finally, we propose the parameter selection algorithm for this method and make a conclusion of the work.

This work was part of the project HD-Tomo funded by Advanced Grant No. 291405 from the European Research Council.

Lyngby, 31-October-2015

Mikhail Romanov

Acknowledgements

I would like to thank my supervisors, Per Christian Hansen and Anders Bjorholm Dahl for invaluable help in becoming a researcher and introducing me a good culture of writing paper, my parents, Julia Romanova and Victor Efremov and grandparents, Maria Efremova and Alevtina Romanova for a strong psychological support. I would like to thank my girlfriend, Xenia Borisovskaya for patience and understanding and providing an inspiration.

Contents

Chapter 1

Introduction

In which we talk about the reasons of the needs for tomography. This chapter is about the general things in tomography. It concerns such things as why do we need tomography and what are the main applications of tomography. Also in this chapter we will talk about the contribution of our work in this area.

1.1 Tomography: Why Do We Need It?

The world is full of different mechanisms that act in different ways: the trees grow, the engines spin the shafts, the animals breathe, the people make the decisions about their lives, even our planet sometimes produces earthquakes, tsunamies. The question, why the mechanisms, bodies or organisms act in the way they act is one of the most complicated questions of philosophy from ancient times and was referred to as 'The Nature of Things'.

This question is relevant nowadays too: we want to know the principles on which the acting mechanisms or physical bodies are built in order to use these principles to improve the organisation of mechanisms, heal the human beings or by other

means to improve our day-to-day life. More than that, this question is far from being closed and answered – if it is possible in principle to answer this question at all.

Some of the principles of the world organisation are not that hard to observe and to research, while the others are quite inconvenient in these terms. The reason is that many of the mechanisms or the objects cannot be cut open. In some cases it is just impossible: for example, it is impossible to cut open the planet or it is quite hard to cut open the mountain, in other cases opening the object or mechanism will violate its internal structure and, thus, it will lose the feature of functioning that we wanted to investigate. As an example, in case one cuts open the engine to see, what happens inside - it will become non-hermetic and will not be able to transform the fuel into the energy. The other very good example in this sense is the human eye: once it is cut open - it cannot function any more. In case one opens the brain - it is not the same brain as before and, although, in principle, it can function after it, in many cases it loses many capabilities.

Thus, we need a technique that does not destroy the completeness of the object, but still can 'look' inside of it. These techniques are called 'Non-Invasive' and 'Non-Intrusive'. The main idea of such techniques is that instead of opening the object we use it as a medium for some physical phenomenon (usually transition, scattering or emission of waves or particles) that affect the object very lightly and does not make harm to its functionality and internal structure or as a source.

The data about how the object transmits, scatters or emits in different directions (and, in some cases, in different conditions) is recorded and after that is used to compute the model of the object in hand. This process is known as reconstruction. After the reconstruction is done, we have a notion about how the object looks inside and how it functions or why it does not function as it should, while the object itself is still complete and not torn apart.

1.2 Tomography Applications

The tomography may be found not in many areas of our life, while these areas are very important for the quality of our existence.

Medicine: This is the number one application of tomography. The investigating what is wrong with a patient is an important question and it is better in case the answer may be found without breaking into the body of the patient and risking

his life.

In this area X-ray tomography is usually used to investigate bones, lungs, stomach, teeth, breast, intestine, etc. This is one of the cheapest and most commonly used tomographic techniques in medicine.

Another quite common approach in this area is Ultrasound investigation. It is very common to use it in liver diagnosis, pancreas, kidneys, heart and many other organs. This technique is also extremely cheap. Also, it is harmless for the body. On the other hand, the precision of this approach is quite low.

One more approach to medical imaging is Magnetic Resonance Imaging (MRI) that is used to produce very precise models of the the organs and systems of organs. Although this approach is capable of giving the best results, it is used quite rarely due to its cost. Also, the MRI tomography has some limitations that are connected with usage of very strong magnetic fields in the process of data collection.

Quite rare, but highly developed approach at the moment is Electrical Impedance Tomography. It has quite low precision and is usually used for monitoring.

The top interest at the moment is the problem of deciphering of the human's brain in which tomography plays one of the central roles. Also, a lot of research is done in the area of recognition of the intentions of person in order to provide direct control for some devices to the brain.

Geological Survey is another important application of tomography. The main role of tomography here is to find mineral resources as well as to investigate the structure of the subsurface. Due to the high cost of drilling the survey is very important in these cases.

Usually the data collection in this case is based on so called Seismic Tomography, where several sources of seismic waves (usually created by explosions) produces strong acoustic waves that afterwards are detected by seismographs in different locations. Because we have a very limited access to the interior of the Earth, the problem of reconstruction of the volume of interest is quite a hard task.

Another application of tomography in this area is the survey of the ocean's bottom. In this case devices that are called echo-sounders are used. It is also important for the extraction of mineral resources from the sea bottom and, especially, for extraction of oil and natural gas from the shelf.

Engineering. This application area is very important for the industrial purposes. Usually before the mechanism or the material is being built, many numerical and physical experiments are carried out. Unfortunately, these experiments have in many cases serious limitations and, due to that, there is a need to verify the declared features of the product.

There are many specific applications in this area. One of the most complicated tasks here is the scanning of engines and other mechanical devices. The problem is that usual measurement instruments are not capable of penetrating the metal parts - and, thus, it is hard to make a good scan of how the gas flows inside of the engine tubes and what happens inside of cylinders. Fortunately, there is a way to scan such devices – the neutron tomography. Unfortunately, the equipment that is needed for neutron tomography is quite large, that makes this technique quite expensive and less popular than the X-ray scanners.

Another application concerns the research of building materials [2], [4], [3]. In these cases the Electrical Impedance Tomography is used as it is the best candidate for detection of the conductor disconnection. Usually the fragment of the material is exposed to extremal overloads and the interior structure is observed.

To sum up, it is obvious that the tomography is highly relevant field in science, healthcare and engineering and it is worth looking into it and improving it.

More about the applications and instruments of tomography may be found in the book [1].

1.3 Research Problem Formulation

It is quite logical that the more data we collect about the object of interest – the more we know about the object – the better is the reconstruction of this object. Also, big amounts of data may compensate the imperfectness of the collected data: as it is quite hard to make a perfect measurements, usually the data is measured with some errors or noise that may be produced by the physics of the process.

In many cases the collection of big amounts of data is unwanted due to cost of the measurements, the exposure to X-rays, time of making measurements or many other reasons. Thus, there is a need for compensating for the lack of data by some other knowledge about the object of interest. The more we know about

the object, the more data we can compensate. Here it is important to note that we should use only the information that is relevant for the object. Otherwise we may get a reconstruction that has nothing in common with the real object.

In this work we concentrate on CT, and we investigate if we can use the knowledge about the materials (we assume that we know the attenuation coefficients of the materials and information is given beforehand in the form of mean value of the attenuation coefficient and standard deviation of the attenuation coefficient) that the object consists of. That seems to be a good idea as it is a strong knowledge about the object: since we know the materials the only problem is now to locate the positions of the material clusters. This problem should be simplier than the problem of computing whole image.

1.4 Thesis Organisation

We start with the chapter where we overview, how the measurements are carried out. As the main objective of our work is the Transmission Tomography problem, in this chapter we carefully discuss X-Ray tomography, but also we take a quick look at the Emission Tomography problem.

After that we consider the aspects of the geometry of the measurement devices, which of the geometries are used in different devices.

Next, we discuss the classical reconstruction algorithms and segmentation algorithms, then we do the overview fo relevant work, where the reconstruction and segmentation are combined into one procedure.

Then we give exact copies of our paper in Chapter 6 that was submitted to the journal "Inverse Problems in Science and Engineering" in coauthorship with Per Christian Hansen, Ander Bjorholm Dahl and Yiqiu Dong. We have also done two technical reports the texsts of which we give in Chapters 7 and 8: "Relaxed Simultaneous Tomographic Reconstruction and Segmentation with Class Priors for Poisson Noise" and "A Parameter Choice Method for Simultaneous Reconstruction and Segmentation".

Bibliography

[1] John Banhart. "Advanced tomographic methods in materials research and engineering". In: (2008).

[2] Tsung-Chin Hou and Jerome P Lynch. "Electrical impedance tomographic methods for sensing strain fields and crack damage in cementitious structures". In: *Journal of Intelligent Material Systems and Structures* (2008).

[3] Kimmo Karhunen et al. "Electrical resistance tomography imaging of concrete". In: *Cement and Concrete Research* 40.1 (2010), pp. 137–145.

[4] R Lazarovitch, D Rittel, and I Bucher. "Experimental crack identification using electrical impedance tomography". In: *NDT & E International* 35.5 (2002), pp. 301–316.

Chapter 2

Measurement Techniques

In this chapter we consider more closely the means of measurements that are used in tomography and that are relevant for our work. We also take a brief look at the geometry of measurements as it is an important aspect of the measurement process.

The presentation in this chapter is based on the book [1].

2.1 X-Ray and Transmission Tomography with Applications

The most common way to look into an object is based on the technology of X-ray scanning. The principle of the measurements is based on the high energy photons that propagate through the object along some trajectory, but as the photons have quite small wavelengths, it is possible to neglect the deviations from the straight line. The photons are usually created by a high energy photon source and after that it is usually collimated to avoid the unwanted background radiation. The

majority of photons propagate along the directions that are not limited by the collimator. Photons, after penetrating the object, are caught by the detector which counts the photons that have reached it.

The photon source is a vacuum tube that contains cathode and anode. The photon source works according to the following principle: the electrons leave the cathode and are accelerated by an electric field. Then the accelerated electrons collide with the anode. When this collision happens, the electron looses the majority of its energy in the form of radiation. This radiation forms the X-ray beam.

As this process is driven by a big amount of particles, the nature of the process is statistical. That means that in different moments of the process a different amount of photons are being produced. It is possible to show that the amount of the photons n that are being produced in one time unit follows a Poisson distribution:

$$p(\mathcal{N}_e = n) = \frac{(\mathbf{E}(\mathcal{N}_e))^n}{n!} \exp(-\mathbf{E}(\mathcal{N}_e)), \tag{2.1}$$

where $\mathbf{E}(\mathcal{N}_e)$ is an expected amount of emitted photons (that can be treated as the average amount of emitted photons per time unit over the long time).

As the photons propagate along a straight line, some of the photons interact with the medium through which they propagate. Because of these interactions not all of the photons that were emitted by the source in the direction of the detector reach the destination.

The interaction of the photons with the medium may be described as the superposition of two physical phenomena: one is the refraction of the photons, another is the absorption. In both cases the photons are deviated form movement along the straight line. As we can detect only the amount of photons that were not deviated from the straight line, the interaction of the X-ray photons with the material is characterized by the material-specific attenuation coefficient that takes into account both refraction and absorption. The important part here is that the attenuation coefficient depends only on the material (and is completely defined by the properties of the material) and does not depend on the location of the material or the time when the interactions take place.

In case the photon passes through a small piece of material with attenuation coefficient μ and the length of the ray inside of the material is dl, then the photon has the following probability of passing through:

$$\mathcal{P}_{tl} = 1 - \mu_l \, dl_l. \tag{2.2}$$

Taking into account that the probability of passing through the system of pieces of materials is equal to the product of probabilities:

$$\mathcal{P}_t = \prod_l \mathcal{P}_{ti} dx. \tag{2.3}$$

Taking the logarithm of this expression it is possible to convert the product into the sum:

$$\log \mathcal{P}_t = \sum_l \log \mathcal{P}_{tl} \tag{2.4}$$

The logarithm of the equation (2.2), taking into account that $dl \to 0$, turns into the following expression:

$$d \log \mathcal{P}_{tl} = \log(1 - \mu_l \ dl_l) = -\mu \ dl. \tag{2.5}$$

In the real world we deal with objects with finite length and the attenuation coefficient in different places of the object may be different (as some of the places may be made of different materials). The overall probability of passing through is the product of probabilities of passing through the small slices, or the logarithm of the probability of passing through the object is the sum of logarithm of probabilities of passing through a small slices of object:

$$d \log \mathcal{P}_t = -\int \mu(l) \ dl. \tag{2.6}$$

The expected value of the transmitted photons is

$$\mathbf{E}(n_r) = \mathcal{N}_e \mathcal{P}_t. \tag{2.7}$$

Thus, substituting P_t in (2.7) by (2.6), we get

$$\mathbf{E}(\mathcal{N}_t) = \mathcal{N}_e \exp\left(-\int \mu(l) dl\right). \tag{2.8}$$

As each of the photons carries some amount of energy, we may conclude that the intensity of the X-ray after passing through the object is equal to

$$I = I(0) \exp\left(-\int \mu(l) dl\right), \tag{2.9}$$

where I(0) is the intensity of the X-Ray before the object. This relation is known as Lambert-Beer's Law.

As in the case with photon emission, the process in this case has a statistical nature (many of the photons pass through the object) – that means that in the

different moments different amount of photons passes through. The probability that n_t photons are transferred through the object may be represented by the Bernoulli distribution given that the source emitted $\mathcal{N}_e$ photons:

$$\mathcal{P}(\mathcal{N}_t = n|\mathcal{N}_e) = \binom{\mathcal{N}_e}{n}\mathcal{P}_t^n(1-\mathcal{P}_t)^{\mathcal{N}_e-n}. \tag{2.10}$$

It is possible to show that in case we assume that the $\mathcal{N}_e$ is distributed according to (2.1) - then the overall probability of the transmitted photons obeys the distribution:

$$\mathcal{P}(\mathcal{N}_t = n) = \frac{(\mathcal{P}_t\mathbf{E}(\mathcal{N}_e))^n}{n!}\exp\left(-P_t\mathbf{E}(\mathcal{N}_e)\right). \tag{2.11}$$

As we see, the amount of transmitted photons is also distributed according to a Poisson distribution, where the parameter of the distribution has changed from $\mathbf{E}(\mathcal{N}_e)$ in (2.1) to $\mathcal{P}_t\mathbf{E}(\mathcal{N}_e)$ (2.11).

After the photons pass through the object they collide with a detector. The detector may with some probability $\mathcal{P}_d$ detect a photon, while the others are not detected by it. The probability distribution of detection of n photons is described by the following expression:

$$\mathcal{P}(\mathcal{N}_d = n|\mathcal{N}_t) = \binom{\mathcal{N}_t}{n}\mathcal{P}_d^n(1-\mathcal{P}_d)^{\mathcal{N}_t-n}. \tag{2.12}$$

Again, as in the situation with transmission, it is possible to show that in case we take into account the distribution of the transmitted photons in (2.11), the overall probability of detection of $\mathcal{N}_d$ photons is equal to

$$\mathcal{P}(\mathcal{N}_d = n) = \frac{(\mathcal{P}_d\mathcal{P}_t\mathbf{E}(\mathcal{N}_e))^n}{n!}\exp\left(-\mathcal{P}_d\mathcal{P}_t\mathbf{E}(\mathcal{N}_e)\right). \tag{2.13}$$

Again, the amount of detected photons is distributed according to a Poisson distribution. Here the parameter of the Poisson distribution is equal to $\mathcal{P}_d\mathcal{P}_t\mathbf{E}(\mathcal{N}_e)$.

When the amount of emitted photons is very large (as it usually is), one can derive from (2.13) the following expression:

$$\mathcal{P}(\mathcal{N}_d = n) = \frac{1}{\sqrt{2\pi}\mathcal{P}_d\mathcal{P}_t\mathbf{E}(\mathcal{N}_e)}\exp\left(-\frac{(\mathcal{P}_d\mathcal{P}_t\mathbf{E}(\mathcal{N}_e)-n)^2}{\mathbf{E}(\mathcal{N}_e)}\right). \tag{2.14}$$

As we can see, when $\mathcal{N}_e \to \infty$, the amount of detected photons is distributed according to a Gaussian distribution with mean equal to $\mathcal{P}_d\mathcal{P}_t\mathbf{E}(\mathcal{N}_e)$ and a standard deviation $\sqrt{\mathcal{P}_d\mathcal{P}_t\mathbf{E}(\mathcal{N}_e)}$.

It is important to note that, according to this result, the signal-to-noise-ratio is equal to

$$SNR = \frac{\mathbf{E}(\mathcal{N}_e)}{\sqrt{\mathbf{E}(\mathcal{N}_e)}} = \sqrt{\mathbf{E}(\mathcal{N}_e)}. \tag{2.15}$$

Thus, the more photons we detect, the more reliable is the result. On the other hand, as the attenuated photons affect the object of interest, in many cases it is not wanted to expose the object to big amount of X-ray photons. Excess of exposure of the object to the X-ray radiation may result in damage of the internal structure of the object. In case of medical this may lead to cancer, in case of material study this may result in change of chemical composition of the object. Thus, one should find a compromise between the reliability of data and the damage to the object.

To sum up, the process of measurements of X-ray photons has a statistical nature. Due to this it is hard to make the precise measurements of the integral of the attenuation coefficient $\int \mu(l)dl$ and the data that we use to make our recononstructions from the projections contains noise. In case we do low-dose X-ray tomography, this noise is distributed according to a Posson distribution. Otherwise we can approximate Poisson distribution with Gaussan distribution according to the (2.14). We will use this result below.

2.2 Emission Tomography with Applications

Emission tomography has several important differences compared to X-ray tomography. The most important is that in case of Emission Tomography the role of the source of the photons plays the object itself. Because of that the probabilities of recording the photons are described by different expressions.

First of all, in this case instead of the attenuation coefficient μ, the activity γ of different areas is what we want to find. The object generates the photons, and then the overall amount of photons is measured in the end.

Consider the process of measuring of the amount of photons that were emitted in the direction of the detector by small pieces of object that are situated on the line l (we can catch only the photons that were emitted by these pieces by inserting collimator in front of the detector). The expected value of photons that the small piece of object emits in each direction is the same and along the line l is $\gamma(l)dl$. The overall expected amount of emitted photons by the pieces of object on the

line l is equal to

$$\mathbf{E}(\mathcal{N}_e) = \int \gamma(l)dl. \tag{2.16}$$

The overall amount of emitted photons along the line l in the direction of the detector is distributed according to a Poisson distribution:

$$\mathcal{P}(\mathcal{N}_e = n) = \frac{(\mathbf{E}(\mathcal{N}_e))^n}{n!} \exp\left(-\mathbf{E}(\mathcal{N}_e)\right). \tag{2.17}$$

Thus, by analogy with X-ray scanning, we can conclude that the overall amount of detected photons is equal to

$$\mathcal{P}(\mathcal{N}_d = n) = \frac{(\mathcal{P}_d\mathbf{E}(\mathcal{N}_e))^n}{n!} \exp\left(-\mathcal{P}_d\mathbf{E}(\mathcal{N}_e)\right). \tag{2.18}$$

As we can see from equation (2.18), the probability is slightly different from equation (2.13). This especially concerns the term $\mathbf{E}(\mathcal{N}_e)$ which has significantly different form.

2.3 On the Geometry of Measurements

As we have already mentioned, we consider only cases where the measurements are made along straight lines. The location of these straight lines is referred to as the geometry of rays. The geometry of rays vary from problem to problem and depends on the locations of the source of photons and the detectors.

The easiest geometry for the reconstruction is a parallel-beam projection. The data that was generated with the help of this approach is very easy to deal with by means of the very traditional and most commonly used algorithms in the area: the Filtered Back Projection (FBP). The shortcoming of this geometry is that the parallel beam is quite hard to get: either the source and the detector should be able to move along the parallel rails or there should be many sources that emit parallel beams of photons independently. The problem of the first approach is that the time of measurements is quite big in this case as in between making measurements for the same projection we have to move the source and the detector physically and that takes the majority of time of the measurements. Also, as the mechanical details are not perfect, the errors in the positions of projections are quite often and the devices do not last long. The problem of the second approach is that we need several sources of the rays, one for each projections. That makes the devices quite expensive. In both cases the system

physically rotates around the object that also creates similar inconveniences as with moving the detector and the source.

More practical approach is to use the fan-beam geometry for the medical devices. In these kind of devices only one source of the X-Rays is used and the array of the detectors records the photons that were emerged by the only source in several directions. This approach of measurements is more convenient than the parallel-beam geometry - it is cheap, more robust and the speed of measurements is higher. The Filtered Back Projection algorithm with some modifications [3] may be applied to reconstruction of the object interior from this data too. Nevertheless, this system still has problems: still whole system rotates around the object.

Even more advanced approach is instead of using the physically rotating system is to use the electrically controlled beam of electrons that falls on the round anode that is situated around the object of interest (Electron Beam Tomography) [2]. The anode emits the photons that make projections. These devices make very fast projections. Although the geometry of these devices is fan-beam, the different beam may lay in different planes - this makes the geometry even more complicated.

In many cases the object is scanned and reconstructed slice-by-slice - and, thus, the 3D object interior is reconstructed as a set of 2D images. It is acceptable in many cases, but to reduce the amount of projections it is often better to make the projections in more sophisticated manner. Quite often the helix trajectory of projections is used, when the plane next projection is slightly shifted from the previous one. In this case it is not possible to compute the 3D object as a set of 2D slices.

Bibliography

[1] Thorsten M Buzug. *Computed tomography: from photon statistics to modern cone-beam CT*. Springer Science & Business Media, 2008.

[2] *Electron beam tomography*. `https://en.wikipedia.org/wiki/Electron_beam_tomography`. [Online; accessed 21-November-2015].

[3] Heang K Tuy. "An inversion formula for cone-beam reconstruction". In: *SIAM Journal on Applied Mathematics* 43.3 (1983), pp. 546–552.

Chapter 3

Classical Methods of Reconstruction

There is a variety of methods of reconstruction of the object model from data. In this chapter we will make an overview of the most used techniques for reconstruction. We will consider their strong and weak sides and peculiarities in a brief way.

3.1 Inverse Radon Transform: Filtered Back Projection

The Filtered Back Projection (FBP) is the most common reconstruction technique that is widely used in the industry due to its simplicity and speed. We will not consider this technique in details here, but we will say a couple of words about it's features.

This technique deals with data that is generated by the Radon Transform that is usually known as a sinogram. It is possible to show that in case we have a full set

of data (that means that we have all the projections and all the values of rays), we may fully reconstruct the object using the Filtered Back Projection. Unfortunately, it is not possible to collect all the data as the amount measurements that should be done is infinite. Thus, we have to deal with the fact that sometimes the image will not be reconstructed perfectly in all the cases. More than that, as to make the perfect measurements is impossible, the reconstruction of the object will also be affected by the noise.

As for the shortcomings of the FBP, the main drawback is that to get a reliable reconstruction a lot of data is needed. Besides that FBP produces the blurry results, with low amount of data it tends to produce the artifacts and its inputs be preprocessed heavily in many cases.

More information about the FBP may be found in [17], [4], [22], [26].

3.2 Algebraic Reconstruction

This chapter is based on the software package [14].

In the previous section we considered a problem of reconstructing an object from the data, where the data was obtained as a continuous transform from the image to the sinogram. As it is impossible to measure all the data it is possible to consider the discretization of the sinogram and the image may be considered as a set of pixels that has finite size and inside of these pixels the attenuation coefficient is uniform.

In this chapter we will consider the image $\mathbf{x}$ to be a set of pixels, where each of the pixles of the image j has uniform attenuation coeffitient x_j. Each of the data values b_i is obtained from the image $\mathbf{x}$ using the following rule:

$$b_i = \sum_{j=0}^{N} a_{ij} x_j + \varepsilon_i(b_i), \tag{3.1}$$

where $\varepsilon_i(b_i)$ is an independent noise in the data element with number i and a_{ij} is the length of the ray along which the measurement i is made inside of the pixel j. Also, we substitute the integral with sum as we are dealing with discrete model of object. We should note here that the noise models were described in chapter 2. Also we should note that, in general, the noise level may depend on different things, but in our work we consider only the Poisson and the Gaussian noise.

The Poisson noise depends only on the signal value, while the Gaussian noise does not depend on the signal value. We also should note that we consider that the noise in different data instances is independent. The values a_{ij} are defined by the geometry of measurements.

This equation may be reformulated in a more convenient matrix notation:

$$\mathbf{b} = A\mathbf{x} + \varepsilon(\mathbf{b}). \tag{3.2}$$

The Algebraic Reconstruction approaches try to solve the problem without respect to the noise. In these approaches we consider that

$$\mathbf{b} = A\mathbf{x} \tag{3.3}$$

and we try to solve the problem of finding image from data as a problem of solving a system of ordinary linear equations.

More information about algebraic reconstruction techniques may be found in books [12], [19].

3.2.1 Pseudoinverse Matrix

As we want to find a solution to the system of linear equations, it may seem logical to solve it using the inversion of the matrix A:

$$\mathbf{x} = A^{-1}\mathbf{b}. \tag{3.4}$$

Unfortunately, this is only possible in case the matrix A is a square and full-rank matrix. In many problems we have less data points than pixels in image (in this case the problem is called underdetermined) and we have to deal with this or we may have more data then the amount of pixels (in this case the problem is called overdetermined) and we would like to use it to be able to produce a better result.

One of the ways to solve this problems is instead of using the inverse matrix A^{-1} to use the pseudoinverse matrix A^{+}. Then the solution to the problem may be found in the following way:

$$\mathbf{x} \approx A^{\dagger}\mathbf{b}. \tag{3.5}$$

The pseudo-inverse matrix was presented by Moore [23] and Penrose [27]. In case the matrix A is a full-rank matrix and has linearly independent columns, the pseudoinverse is equal to

$$A^{\dagger} = \left(A^T A\right)^{-1} A^T. \tag{3.6}$$

In case the matrix A is a full-rank matrix and has linearly independent rows, the pseudoinverse is equal to

$$A^{\dagger} = A^T \left(AA^T\right)^{-1}. \tag{3.7}$$

There are many ways to compute the pseudoinverse matrix and we do not consider them here.

We should note that solving the problem using the pseudoinverse matrix is not reliable as in case the data contains some amount of noise and is underdetermined - the results usually contain a lot of noise. Thus, in case we have noise in data we have to compensate it with bigger amount of data.

3.2.2 Landweber Iteration

As in the ideal case the image $\mathbf{x}$ satisfies the equation (3.3), this also means that

$$A\mathbf{x} - \mathbf{b} = 0. \tag{3.8}$$

Thus, the problem of solving the (3.3) is equivalent to finding the minimum of the squared residual function, i.e.

$$\arg\min_{\mathbf{x}} \phi(\mathbf{x}) = \arg\min_{\mathbf{x}} \frac{1}{2}(A\mathbf{x} - \mathbf{b})^T(A\mathbf{x} - \mathbf{b}). \tag{3.9}$$

That approach is referred to as least squares problem. We should note here that the system (3.3) may be unsolvable, but we may consider the solution to the least squares problem to be the alternative to the solution as it is the solution $\mathbf{x}$ that minimizes the residual (3.3).

To solve the least squares problem (3.9) we may use the standard gradient descent approach: we start with some initial approximation $\mathbf{x}^0$. Then we do the update in the following way:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla\phi(\mathbf{x}^k). \tag{3.10}$$

It is easy to get the following expression for the gradient of the least squares function:

$$\nabla\phi(\mathbf{x}^k) = A^T(A\mathbf{x} - \mathbf{b}). \tag{3.11}$$

Thus, the gradient descent transforms into the following rule:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k A^T(A\mathbf{x} - \mathbf{b}), \tag{3.12}$$

where α_k is a step length, k is a number iterations that already been done. This method was first presented in [21].

It is worth noting that using this technique it is guaranteed that we find the minimum of the least squares problem with any precision given beforehand (unless it is smaller than the machine epsilon for a chosen precision). This is due to the problem (3.9) is convex.

There are several approaches that may be treated as a modification of this approach. One is the Cimmino iteration, where the next approximation is computed the following expression:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k A^T \frac{1}{m} \text{diag}\left(\frac{1}{\sum_{j=0}^{N} a_{ij}^2}\right)\left(\mathbf{b} - A\mathbf{x}^k\right). \tag{3.13}$$

Another important approach is known as Simultaneous Algebraic Reconstruction Technique (SART) [1]. The next iterate in this approach is computed in the following way:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \text{diag}\left(\frac{1}{\sum_{j=0}^{N} a_{ij}}\right) A^T \left(\frac{A\mathbf{x} - \mathbf{b}}{\sum_{i=0}^{M} a_{ij}}\right). \tag{3.14}$$

There are many other flavours of the algorithms above, but the ones that were presented above are the most important ones.

3.2.3 Row Action Methods

In this section we will consider a special flavour of Algebraic Methods which, instead of using the whole matrix of the linrear system A, uses instead only one of the rows of the matrix $a_{i:}$ to compute the next iterate.

These methods also may be considered as the methods that solve the problem (3.9) in the following way: the minimization problem is considered as a sum

$$\arg\min_{\mathbf{x}} \|A\mathbf{x} - \mathbf{b}\|_2^2 = \arg\min_{\mathbf{x}} \sum_{i=0}^{M} (a_{i:}\mathbf{x} - b_i)^2. \tag{3.15}$$

Each iteration minimizes one of the sum's components. To decrease the component with index i, the new image $\mathbf{x}^{k+1}$ should be closer to the projection of the current solution $\mathbf{x}^k$ on the surface that is defined by the matrix's row $a_{i:}$ than the current solution. The direction of the projection is defined by the vector

the elements of which are the values in the column $a_{i:}$. Thus, the new image is computed according to the following rule:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha a_{i:}, \tag{3.16}$$

where α is a step length. The distance from the current point $\mathbf{x}^k$ to the plane defined by $a_{i:}$ may be shown to be equal to

$$\rho = \frac{1}{\|a_{i:}\|_2} \left(a_{i:}\mathbf{x}^k - b_i \right). \tag{3.17}$$

Thus, the new image may be computed by the following rule:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha' \frac{b_i - a_{i:}\mathbf{x}^k}{||a_{i:}||_2^2} a_{i:}, \tag{3.18}$$

where $\alpha' \in (0, 1]$ is a relaxation parameter that shows how close should we be to the projection of $\mathbf{x}^k$ on the plane defined by $a_{a:}$. This algorithm is called a Kaczmarz's iteration [18].

This algorithm has several important features that we have to mention here: first, in case the iterations converge to the least squares solution in case the $\alpha' < 1$. Besides that, in case the object has sharp edges, Kaczmarz's method has the feature that is called semiconvergence [6]: the best aproximation to the solution in terms of the second norm $\frac{\|\mathbf{x}^k - \mathbf{x}\|_2}{\|\mathbf{x}\|_2}$ is obtained after a finite amount of iterations, but after that the algorithm starts to overfit the noise in the data. Thus, it is possible to get better solution than the least squares solution just by stopping the iterations at the particular point (this approach is called Early Stopping [33]).

3.2.4 SVD

Another approach to solve the problem (3.9) is to use the Singular Value Decomposition (SVD) [11].

The SVD decomposition of the matrix is defined as follows:

$$A = U\Sigma V^T, \tag{3.19}$$

where U and V are the unitary matrices, Σ is a diagonal matrix that consists of the so-called singular values. The matrices U, V and singular values have the following feature:

$$\sigma_l u_{l:} = A v_{l:}, \tag{3.20}$$

and

$$\sigma_l^{-1} v_{l:} = A^{-1} u_{l:} \tag{3.21}$$

where σ_l is the i-th singular value, the vectors $u_{i:}$ and $v_{i:}$ are the corresponding rows from the matrices U and V respectively. Usually. for convenience, the singular values in the diagonal of the matrix Σ are ordered descending.

We should note here that the matrices U and V have the following feauture:

$$\mathbf{x} = VV^T x = \sum_l v_{l:}^T \mathbf{x}\ v_{l:} \tag{3.22}$$

and

$$\mathbf{b} = UU^T u = \sum_l u_l^T \mathbf{b}\ u_l, \tag{3.23}$$

where index l runs over all of the rows. Thus, the matrix's U rows may be considered to be a basis in the space of the image $\mathbf{x}$, while the matrix's V rows may be considered to be a basis in the space of the data $\mathbf{b}$.

Using this matrix decomposition (3.23) and the relation (3.21), we may constuct the solution in the following way:

$$\mathbf{x} = A^{-1}\mathbf{b} = \sum_l \frac{u_{l:}^T b}{\sigma_l} v_{l:}. \tag{3.24}$$

The problem of inverting the matrix A as shown in (3.24) may cause problems as the singular values σ_l decrease to 0 with growth of i, but due to the presence of noise in the data $\mathbf{b}$, the numerator after some time stops tending to zero (Picard Condition [13]). Because of this in the expression (3.24) with growth of l the ratio may start to tend to infinity, overfitting the noise and spoiling the solution.

One of the ways to deal with this problem may be solved as in the previous chapter: we may only leave several first v_l that will go to our solution and leave out all the others (Truncated SVD [32], [15]). Another way to avoid this problem is to substitute in the denominator the σ_l with value

$$\left(\frac{\sigma_l}{\sigma_l^2 + \lambda^2} \right)^{-1}, \tag{3.25}$$

where $\lambda > 0$ and thus, we will prevent the divisions by zero in the expression (3.24). As a result, the expression (3.24) will transform to

$$\sum_l \frac{u_{l:}^T b \sigma_l}{\sigma_l^2 + \lambda^2} v_{l:}. \tag{3.26}$$

This approach may be referred to as Tikhonov Regularization. Changing the parameter λ we may regulate, how much the solution will be affected by the rows of the matrix V that correspond to the low singular values.

3.3 Variational and Statistical Methods

In the previous section we have already considered the problem of reconstructing the object $\mathbf{x}$ from the data $\mathbf{b}$ as a minimization problem. In this chapter we will consider how to formulate the minimization problem and discuss the general techniques to solve this minimization problem.

As was already stated above, the problem of finding the solution to the system of linear equations

$$A\mathbf{x} = \mathbf{b} \tag{3.27}$$

may be represented as a minimization problem

$$\arg\min_{\mathbf{x}} \|A\mathbf{x} - \mathbf{b}\|_2^2. \tag{3.28}$$

It is possible to show that these two formulations are equivalent to each other: to find the stationary point of (3.28) one needs to take the gradient of the minimized expression and after that should assign it to zero and solve the problem. The minimized function in (3.28) may be represented in the following way:

$$\|A\mathbf{x} - \mathbf{b}\|_2^2 = (A\mathbf{x} - \mathbf{b})^T (A\mathbf{x} - \mathbf{b}). \tag{3.29}$$

Taking the gradient of this and assigning it to zero we get the following equation:

$$2A^T(A\mathbf{x} - \mathbf{b}) = 0 \tag{3.30}$$

that is equivalent to the linear system (3.27).

There are many ways to solve the minimization problems. This includes simple methods as gradient descent, more complicated methods as Newton methods (it is not recommended to use such methods here as the Hessian matrix has $O(N)$ elements, thus, the time that is needed to estimate it will be quadratic), Quasi-Newton methods such as BFGS [3], [8], [10], [29], and more complicated - as Conjugate Gradients method [7], Limited BFGS (L-BFGS) method [25], [5], Nesterov-type methods [24] and many others.

Unfortunately, in case the matrix A has less rows than columns - that means that we have less data than variables. Thus, in this case, the solution of the system is non-unique (and that means that there are many minima of the minimized function). The presence of noise in the data $\mathbf{b}$ may spoil the situation even more: it may drive the solution of the system of linear equations far from the true solution. To deal with that we need to improve the minimization problem. To make the solution of the minimization problem more robust to the noise, we introduce the regularization:

$$\|A\mathbf{x} - \mathbf{b}\|_2^2 + \lambda \mathrm{Reg}(\mathbf{x}), \tag{3.31}$$

where Reg is a function that penalizes some unwanted features of the image. The λ is a regularization parameter and it regulates, how much influence the regularization has on the solution. Many regularization functions may be used here to make the solution more robust. The classical regularization term is

$$\mathrm{Reg}(\mathbf{x}) = \|\mathbf{x}\|_2^2. \tag{3.32}$$

It is also known as Tikhonov regularization ([31], [30]). It penalizes the grey levels of the pixels of the image – and, thus, prevents them from getting very high values. Another quite popular regularization term is

$$\mathrm{Reg}(\mathbf{x}) = \sum_j \|D_j\mathbf{x}\|_2^2, \tag{3.33}$$

where D_j is the discrete approximation of the gradient in the pixel j. This regularization term penalizes the gradient of the image – thus we prefer the images with less gradient to the images with more gradient.

Another very important type of regularization is called Total Variation (TV) regularization [28]. The Total Variation corresponds to the following regularization term:

$$\mathrm{Reg}(\mathbf{x}) = \sum_j \|D_j\mathbf{x}\|_2. \tag{3.34}$$

One of its main features is that it prefers the reconstructions with sharp edges. Another feature is that it penalizes the gradients. Thus, this regularization term is used when one wants to reconstruct the sharp objects from the noisy data. Also, this regularization approach is used in image deblurring when we have objects that are almost constant inside and few sharp edges of the objects.

We should note here that all the regularization parameters that were introduced above keep the problem (3.31) convex. Also it is possible to show that all these problems with regularization parameter $\lambda > 0$ have the only solution. It is quite

well-known that these regularization techniques treat the noise well and are also capable of dealing with insufficient data.

The formulation (3.31) may be obtained using the Maximum a Posterioiri Probability approach (MAP). Consider the problem of reconstructing the image $\mathbf{x}$ from the data $\mathbf{b}$, where the data is noisy and we know the statistics of this noise. We may formulate this problem in the following way:

$$\arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{b}), \tag{3.35}$$

where $p(\mathbf{x}|\mathbf{b})$ is a probability of image $\mathbf{x}$ given the fixed data $\mathbf{b}$. This formulation of reconstruction problem is quite logical as we want to find the most possible image that given the data that was measured beforehand.

We may modify this problem according to Bayes rule:

$$\arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{b}) = \arg\max_{\mathbf{x}} \frac{p(\mathbf{b}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{b})}. \tag{3.36}$$

In the formulation (3.36) we may see in the enumerator the product of probabilities of obtaining data $\mathbf{b}$ given the image $\mathbf{x}$ and the apriori probability of image $\mathbf{x}$. The first term $p(\mathbf{b}|\mathbf{x})$ corresponds to the noise model of the system and is also often referred to as data fitting term or data fidelity term. The second term represents the probability of the image itself. For example, we may think of the image that has too big values of the gradient to be less possible than the image with lower values of gradient. The value in the denominator is constant in terms of $\mathbf{x}$ and, thus, can not be minimized.

The formulation (3.36) is a product of probabilities. It is possible to show that the maximum of the positive function is equal to the maximum of the logarithm of this function. Knowing this and the fact that the logarithm turns products to sums it is possible to reformulate the problem (3.36):

$$\arg\max_{\mathbf{x}} \frac{p(\mathbf{b}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{b})} = \arg\max_{\mathbf{x}} \log p(\mathbf{b}|\mathbf{x}) + \log p(\mathbf{x}) + \text{const}. \tag{3.37}$$

Now we may formulate each of the terms of the minimization problem. In case the noise is Gaussian, indepent in each of the data points, and has the same parameters (i.e., mean value and standard deviation), the data fidelity term may be written in the following way:

$$p(\mathbf{b}|\mathbf{x}) = \prod_{i=0}^{M} \frac{1}{\sqrt{2\pi}\sigma_{noise}} \exp\left(-\frac{(b_i - \sum_{j=0}^{N} a_{ij}x_j)^2}{2{\sigma_{noise}}^2}\right). \tag{3.38}$$

Taking the logarithm of this we will get the expression that is up to a constant coefficient is equal to the function in (3.28):

$$\log p(\mathbf{b}|\mathbf{x}) = \text{const} - \sum_{j=0}^{N} \frac{(b_i - \sum_{j=0}^{N} a_{ij}x_j)^2}{2{\sigma_{noise}}^2}. \tag{3.39}$$

Thus, in case we solve the problem with Gaussian noise it corresponds to the least squares problem (3.28).

We should note here that the noise in the data may be non-Gaussian (the noise models were overviewed in chapter 2), and in this case the data fidelity term will be different and will not resemble the least squares problem. The data fitting term depends on the noise model and the physics of the measurement process.

Now let us consider the term $\log p(\mathbf{x})$. This term could be interpreted as a regularization term. But in this formulation it has a clear meaning: this term represents our knowledge about the image without taking into account any data. This term is also known as prior. We may assign the prior to be equal to any of the regularizations that were introduced above and we may conclude that

$$-\log p(\mathbf{x}) = \lambda \text{Reg}(\mathbf{x}) \tag{3.40}$$

Here we should also note that not all of the priors are convex. Due to this, not for all priors it is possible to solve the problem using the standard optimization techniques. For non-convex priors there exist global optimization techniques such as Monte-Carlo methods [2], [16], Simulated Annealing [20], [9] and many others. Although they are quite powerful as a minimization tools, for the problems with many data it is unwanted to use them as the running time of these methods is exponential.

As an example of the non-convex prior we can give the l_0 norm of the gradient approximation:

$$-\log p(\mathbf{x}) = \lambda \text{Reg}(\mathbf{x}) = \lambda \sum_j I(\|D_j\mathbf{x}\|_2), \tag{3.41}$$

where the function I is an indicator function that is equal to 0 when the argument is equal to zero or is false and is equal to 1 otherwise.

This regularization term penalizes the edges between different grey levels. Thus, the optimizer of the problem with regularization term (3.41) will be more likely

to contain less edges than without this regularization term. Amount of edges will depend on the regularization parameter λ. Unfortunately, this regularization term is non-convex. More than that, it is possible to show that finding the optimizer of the problem (3.31) with regularization term from (3.41) is an NP-hard problem in general.

Bibliography

[1] AH Andersen and Avinash C Kak. "Simultaneous algebraic reconstruction technique (SART): a superior implementation of the ART algorithm". In: *Ultrasonic imaging* 6.1 (1984), pp. 81–94.

[2] Bernd A Berg and Alain Billoire. *Markov Chain Monte Carlo Simulations*. Wiley Online Library, 2008.

[3] Charles George Broyden. "The convergence of a class of double-rank minimization algorithms 1. general considerations". In: *IMA Journal of Applied Mathematics* 6.1 (1970), pp. 76–90.

[4] Thorsten M Buzug. *Computed tomography: from photon statistics to modern cone-beam CT*. Springer Science & Business Media, 2008.

[5] Richard H Byrd et al. "A limited memory algorithm for bound constrained optimization". In: *SIAM Journal on Scientific Computing* 16.5 (1995), pp. 1190–1208.

[6] Tommy Elfving, Per Christian Hansen, and Touraj Nikazad. "Semi-convergence properties of Kaczmarz's method". In: *Inverse Problems* 30.5 (2014), p. 055007.

[7] Reeves Fletcher and Colin M Reeves. "Function minimization by conjugate gradients". In: *The computer journal* 7.2 (1964), pp. 149–154.

[8] Roger Fletcher. "A new approach to variable metric algorithms". In: *The computer journal* 13.3 (1970), pp. 317–322.

[9] William L Goffe, Gary D Ferrier, and John Rogers. "Global optimization of statistical functions with simulated annealing". In: *Journal of Econometrics* 60.1 (1994), pp. 65–99.

[10] Donald Goldfarb. "A family of variable-metric methods derived by variational means". In: *Mathematics of computation* 24.109 (1970), pp. 23–26.

[11] Gene H Golub and Christian Reinsch. "Singular value decomposition and least squares solutions". In: *Numerische mathematik* 14.5 (1970), pp. 403–420.

[12] Per Christian Hansen. *Discrete inverse problems: insight and algorithms*. Vol. 7. Siam, 2010.

[13] Per Christian Hansen. "The discrete Picard condition for discrete ill-posed problems". In: *BIT Numerical Mathematics* 30.4 (1990), pp. 658–672.

[14] Per Christian Hansen and Maria Saxild-Hansen. "AIR tools—a MATLAB package of algebraic iterative reconstruction methods". In: *Journal of Computational and Applied Mathematics* 236.8 (2012), pp. 2167–2178.

[15] Per Christian Hansen, Takashi Sekii, and Hiromoto Shibahashi. "The modified truncated SVD method for regularization in general form". In: *SIAM Journal on Scientific and Statistical Computing* 13.5 (1992), pp. 1142–1150.

[16] W Keith Hastings. "Monte Carlo sampling methods using Markov chains and their applications". In: *Biometrika* 57.1 (1970), pp. 97–109.

[17] Gabor T Herman. *Fundamentals of computerized tomography: image reconstruction from projections*. Springer Science & Business Media, 2009.

[18] Stefan Kaczmarz. "Angenäherte auflösung von systemen linearer gleichungen". In: *Bulletin International de l'Academie Polonaise des Sciences et des Lettres* 35 (1937), pp. 355–357.

[19] Avinash C.. Kak and Malcolm Slaney. *Principles of computerized tomographic imaging*. Society for Industrial and Applied Mathematics, 2001.

[20] Scott Kirkpatrick. "Optimization by simulated annealing: Quantitative studies". In: *Journal of statistical physics* 34.5-6 (1984), pp. 975–986.

[21] Louis Landweber. "An iteration formula for Fredholm integral equations of the first kind". In: *American journal of mathematics* (1951), pp. 615–624.

[22] Günter Lauritsch and Wolfgang H Härer. "Theoretical framework for filtered back projection in tomosynthesis". In: *Medical Imaging'98*. International Society for Optics and Photonics. 1998, pp. 1127–1137.

[23] E. H. Moore. "On the reciprocal of the general algebraic matrix". In: *Bulletin of the American Mathematical Society* 26.9 (1920), 394–395.

[24] Yurii Nesterov. "On an approach to the construction of optimal methods of minimization of smooth convex functions". In: *Ekonomika i Mateaticheskie Metody* 24.3 (1988), pp. 509–517.

[25] Jorge Nocedal. "Updating quasi-Newton matrices with limited storage". In: *Mathematics of computation* 35.151 (1980), pp. 773–782.

[26] Xiaochuan Pan, Emil Y Sidky, and Michael Vannier. "Why do commercial CT scanners still employ traditional, filtered back-projection for image reconstruction?" In: *Inverse problems* 25.12 (2009), p. 123009.

[27] Roger Penrose. "A generalized inverse for matrices". In: *Proceedings of the Cambridge Philosophical Society* 51 (1955), 406–413.

[28] Leonid I Rudin, Stanley Osher, and Emad Fatemi. "Nonlinear total variation based noise removal algorithms". In: *Physica D: Nonlinear Phenomena* 60.1 (1992), pp. 259–268.

[29] David F Shanno. "Conditioning of quasi-Newton methods for function minimization". In: *Mathematics of computation* 24.111 (1970), pp. 647–656.

[30] Andrey Tikhonov. "Solution of incorrectly formulated problems and the regularization method". In: *Soviet Math. Dokl.* Vol. 5. 1963, pp. 1035–1038.

[31] Andrey Nikolayevich Tikhonov. "On the stability of inverse problems". In: *Dokl. Akad. Nauk SSSR*. Vol. 39. 5. 1943, pp. 195–198.

[32] Peiliang Xu. "Truncated SVD methods for discrete linear ill-posed problems". In: *Geophysical Journal International* 135.2 (1998), pp. 505–514.

[33] Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. "On early stopping in gradient descent learning". In: *Constructive Approximation* 26.2 (2007), pp. 289–315.

Chapter 4

Classical Segmentation Techniques

In this chapter we will make a short overview of the segmentation techniques that are related to the segmentation techniques we are using. We analyse pros and cons of these methods.

4.1 Markov Random Field

The Markov Random Fields (MRF) is a quite widely used model for segmentation. It is a discrete model, where each of the pixels (or sometimes superpixels [12]) of the image $\mathbf{x}$ are assigned to one of the possible classes. The information about the classes may be known beforehand or may be extracted from the image using clustering techniques along with segmentation process.

This problem may also be formulated in the statistical way:

$$\mathbf{s} = \arg\max_{\mathbf{s}} p(\mathbf{s}|\mathbf{x}), \tag{4.1}$$

where $\mathbf{s}$ is a segentation of the image $\mathbf{x}$, the elements s_j of the segmentation $\mathbf{s}$ are called the labels. This problem is usually reformulated in terms of a potential:

$$\mathbf{s} = \arg\min_{\mathbf{s}} -p(\mathbf{s}|\mathbf{x}) = \arg\min_{\mathbf{s}} U(\mathbf{s}, \mathbf{x}), \tag{4.2}$$

where U is called a potential and is wanted to be minimized. Then the following assumption is usually made: the potential contains two terms. One of these terms takes into account, how much the segmentation label of a pixel (or superpixel) depends on the features of this pixel (or superpixel). This term is usually referred to as a unary potential. Another term reflects how much the labels of two neighbouring pixels (superpixels) affect each other taking into account the similarity between these pixels. This term is usually referred to as a binary potential.

Thus, the segmentation problem may be reformulated now in the following way:

$$\begin{aligned} \mathbf{s} = \arg\min_{\mathbf{s}} U(\mathbf{s}, \mathbf{x}) = \\ \arg\min_{\mathbf{s}} \sum_j U_u(s_j, x_j) + \sum_j \sum_{j' \in N(j)} I(s_j \neq s_{j'}) U_b(x_j, x_{j'}) \\ \text{s.t.} \quad s_j \in \mathbf{N} \end{aligned} \tag{4.3}$$

where U_u is an unary potential, U_b is a binary potential, j is an index of the pixel (superpixel), $N(j)$ are the indices of neighbours of the pixel (superpixels) of the pixel j.

Usually the first unary potential term takes into account different information about the pixel (superpixel). This may include the colors in the pixel (superpixel), location, in case of the superpixel this may also include different descriptors [5] as well as the area of the superpixel.

The binary potential usually takes into account the similarity between two different pixels/superpixels, but this is not necessary. The binary term plays the same role as a regularization in reconstruction problem. It prevents the segmentation from overfitting the data (that in this case is an image) - and, thus, makes the solution more robust.

The problem (4.3) is a discrete optimization problem. Now we will consider some optimization techniques for these problems.

In case the segmentation contains only two classes (this problem is often referred

to as separation object from the background), it is possible to represent this problem in the form of solving the problem of minimum cut of a graph. (4.3).

This graph may be constructed in the following way: for each of the pixels (or superpixels) of the image we make a vertex. Also we make two vertices one of which is called the source, the other is called the terminal. We connect each of the vertices that correspond to the pixel (or superpixel) j to the terminal with the edge with capacity $U_u(1, x_j)$ and with the source $U_u(0, x_j)$. After this we connect each of the pixels j with each of its neighbors $j' \in N(j)$ with edge with capacity $U_b(x_j, x_{j'})$.

After that the minimum cut algorithm is applied to this graph [1]. The aim of the minimum cut algorithm is to separate the source from the terminal cutting the edges with minimal overall capacities. Thus, when we find the minimum cut we can say that we have found a minimizer to the problem (4.3).

Unfortunately, this only works for the problem of separation of the object from the background. The problem with more than two labels is an NP-hard problem. To find an approximation to the solution of this problem we may use the algorithms that are called Alpha-Expansion algorithm and Alpha-Beta Swaps [4], but these algorithm compute only the approximation.

It is also possible to find an approximation of the problem (4.3) via the approach that is called Message Passing for Markov Random Fields [11].

Other methods that may be applied here are based on the Monte-Carlo method and usually need a lot of time to produce a reliable result.

4.2 Variational l_0 Method

This method, in principle, solves a special case of the problem that was described in section 3.3. In this method we solve the following minimization problem:

$$\begin{aligned} \mathbf{s} = \arg\min_{\mathbf{s}} \quad & \|\mathbf{x} - \mathbf{s}\| + \lambda \sum_j I(\|D_j \mathbf{s}\|) \\ \text{s.t. } & \forall j \; s_j \in \mathbf{R}, \end{aligned} \tag{4.4}$$

where I is an indicator function discussed in equation (3.41), D_j is a discrete approximation of the gradient in pixel j that was introduced in equation (3.33).

The regularization term penalizes the amount of pixels, where the gradient is non-zero. Thus, the amount of jumps of grey levels is forced to be limited. As a consequence, the image that corresponds to the solution of this problem has regions, where the grey level does not change at all. These regions may be called segments, the grey level of the region may be treated as label and the image that corresponds to the solution of this problem may be treated as a segmentation. Thus, we approximate the image $\mathbf{x}$ with another image that is treated as a segmentation $\mathbf{s}$. The data fitting term corresponds to the Euclidean distance between the segmentation $\mathbf{s}$ and the image $\mathbf{x}$ and the amount of jumps in the segmentation is penalized by the regularization term.

In case the segmentation $\mathbf{s}$ contains only two values, this model is referred to as the Chan-Vese model [2]. Also we should note here another important related method of segmentation that also includes the penalty on the curvature of the contour that separates regions with different grey level. It is called the Mumford-Shah model [7].

As was already mentioned above, these models are NP-hard problems.

4.3 Hidden Markov Measure Field Models

The Hidden Markov Measure Field Models (HMMFM) [6] are the relaxation of the concept of the Markov Random Fields on the real number values. The HMMFM assigns to each pixel (or super pixel) j the probability δ_{jk} to belong to the class k. Thus, it is easy to compute the corresponding segmentation to the given HMMFM:

$$s_j = \arg\max_k \delta_{jk}. \tag{4.5}$$

Thus, given a HMMFM there is a unique corresponding segmentation. We may formulate the problem of computing the optimal Hidden Markov Measure Field Model as a minimization problem in case we want to optimize the HMMFM together with class parameters θ in the following way:

$$\begin{aligned} &\arg\max_{\delta,\theta} \frac{p(\mathbf{x}|\delta,\theta)p(\delta)p(\theta)}{p(\mathbf{x})} \\ &\text{s.t. } \forall j \sum_k \delta_{jk} = 1 \\ &\forall j,k \ \delta_{jk} \geq 0, \end{aligned} \tag{4.6}$$

, where $p(\mathbf{x}|\delta,\theta)$ is a marginal probability of image $\mathbf{x}$ given the HMMFM δ and the parameters θ. In case we want to optimize only the HMMFM having the class parameters fixed we may formulate the problem in the following way:

$$\begin{aligned} &\arg\max_{\delta} \frac{p(\mathbf{x}|\delta,\theta)p(\delta)p(\theta)}{p(\mathbf{x})} \\ &\text{s.t. } \forall j \sum_{k} \delta_{jk} = 1, \\ &\forall j,k \ \delta_{jk} \geq 0. \end{aligned} \tag{4.7}$$

We should note here that the constraints come form the definition of the HMMFM: as each of the elements of HMMFM is a probability, it should be greater than zero and overall probability of assigning the pixel j to any class should be equal to 1.

Thus, the problem finding the best HMMFM may be formulated as a optimization problem. This problem may be solved using standard constraint optimization techniques.

4.4 Other Methods

There are many other methods of computing the segmentation of the image, but they are not that related to our work. These methods include watershed method [10], level sets [9], snakes [3], region-growing methods [8] and many others. In our work we deal with the variational formulation - and, thus, it is hard to use those methods.

Bibliography

[1] Yuri Boykov and Vladimir Kolmogorov. "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.9 (2004), pp. 1124–1137.

[2] Tony F Chan, B Yezrielev Sandberg, and Luminita A Vese. "Active contours without edges for vector-valued images". In: *Journal of Visual Communication and Image Representation* 11.2 (2000), pp. 130–141.

[3] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. "Snakes: Active contour models". In: *International journal of computer vision* 1.4 (1988), pp. 321–331.

[4] Herve Lombaert et al. "A multilevel banded graph cuts method for fast image segmentation". In: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. Vol. 1. IEEE. 2005, pp. 259–265.

[5] David G Lowe. "Object recognition from local scale-invariant features". In: *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*. Vol. 2. Ieee. 1999, pp. 1150–1157.

[6] Jose L Marroquin, Edgar Arce Santana, and Salvador Botello. "Hidden Markov measure field models for image segmentation". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25.11 (2003), pp. 1380–1387.

[7] David Mumford and Jayant Shah. "Optimal approximations by piecewise smooth functions and associated variational problems". In: *Communications on pure and applied mathematics* 42.5 (1989), pp. 577–685.

[8] Richard Nock and Frank Nielsen. "Statistical region merging". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.11 (2004), pp. 1452–1458.

[9] Stanley Osher and Nikos Paragios. *Geometric level set methods in imaging, vision, and graphics*. Springer Science & Business Media, 2003.

[10] Jos BTM Roerdink and Arnold Meijster. "The watershed transform: Definitions, algorithms and parallelization strategies". In: *Fundam. Inform.* 41.1-2 (2000), pp. 187–228.

[11] Richard Szeliski et al. "A comparative study of energy minimization methods for markov random fields with smoothness-based priors". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 30.6 (2008), pp. 1068–1080.

[12] Olga Veksler, Yuri Boykov, and Paria Mehrani. "Superpixels and supervoxels in an energy optimization framework". In: *Computer Vision–ECCV 2010*. Springer, 2010, pp. 211–224.

Chapter 5

Joint Reconstruction and Segmentation

Usually the segmentation of the reconstruction is done after the reconstruction has already been computed. The main advantage of this approach is simplicity: the reconstruction process does not need to interact with segmentation process. Thus, it is possible to use standard reconstruction and segmentation techniques. But this approach does not use the information about the classes in the reconstruction process. Thus, it is logical to unite the segmentation and reconstruction into one procedure to make use of prior knowledge about the classes. In this chapter we overview the joint reconstruction and segmentation techniques.

5.1 Simultaneous Reconstruction and Segmentation with Entropy Prior

This approach was presented in [3]. In this paper the problem of reconstructing the image from the CT projections is solved. The reconstruction problem is

formulated using the of Maximum a Posteriori Probability:

$$\arg\max_{\mathbf{x}} \log p(\mathbf{b}|\mathbf{x}) + \log p(\mathbf{x}), \tag{5.1}$$

where $\mathbf{b}$ is the data, the prior $p(\mathbf{x})$ is the exponent of the negative entropy:

$$p(\mathbf{x}) = \exp\left(-\lambda \sum_a \mathcal{P}_{\mathbf{x}}(a) \log \mathcal{P}_{\mathbf{x}}(a)\right), \tag{5.2}$$

where $\mathcal{P}_{\mathbf{x}}(a)$ is a normalized histogram of the image $\mathbf{x}$, i.e., the probability of the randomly chosen pixel to have the grey value a and λ is a regularization parameter.

Unfortunately, this function is non-convex. In this work, the regularization parameter is being increased along iterations to avoid being stuck in local minimum.

This technique does not use segmentation as a special step, but it penalizes the number of different grey levels in the image and, by this, forces the image to have small amount of grey levels. Thus, the value of the grey level may be treated as a label and the image parts that have the same grey level may be treated as the pixels that belong to the same class.

5.2 Discrete Algebraic Reconstruction Technique (DART)

In this section we will consider the Discrete Algebraic Reconstruction Technique (DART) [1]. The DART is a relaxation of the Algebraic Reconstruction Technique on the space of discrete images (i.e., discrete pixel values). DART approach solves the problem that is represented in the form of linear system of equations:

$$\begin{aligned} \mathbf{b} &= A\mathbf{x} \\ \text{s.t. } x_j &\in \mathbf{V}, \end{aligned} \tag{5.3}$$

where $\mathbf{V}$ is a set of allowed grey level values. As was mentioned in Section 3.2, this problem may be considered as a squared residual minimization problem (3.9) and may be solved by means of approaches that were described in sections 3.2 and 3.3. The DART approach is based on the iterative approaches from the section 3.2. To reconstruct the image with this algorithm it is needed to have data $\mathbf{b}$, knowledge

about the geometry in form of matrix A or equivalent and set of discrete values μ_k for the intensity levels known to be present in the reconstructed object.

The algorithm acts in the following way. First, an iterative algorithm (usually SIRT) is run until it yields an approximation for the reconstruction. After that this approximation is thresholded. The thresholds are computed using the following rule:

$$t_k = \frac{\mu_k - \mu_{k+1}}{2}, \tag{5.4}$$

where k is an index of the class. Thresholding produces a new image $\mathbf{x}^\dagger$ where the elements are computed according to the following rule:

$$x_j^\dagger = \begin{cases} \mu_0 & \text{if } x_j < t_0 \\ \mu_1 & \text{if } t_0 \leq x_j < t_1 \\ \dots & \\ \mu_K & \text{if } t_{K-1} \leq x_j \end{cases}. \tag{5.5}$$

After that the boundary pixels are being found. The boundary pixels are the pixels that contain at least two neighbour pixels with different values of the image $\mathbf{x}^\dagger$. The other pixels are called internal pixels (I is a set of the indices of internal pixels).

Then the image is updated in the following way:

$$x_j = \begin{cases} x_j^\dagger & \text{if } j \in I \\ x_j & \text{if } j \notin I. \end{cases} \tag{5.6}$$

After this update is done, the boundary pixels values are recomputed using the iterative algorithm (usually SIRT) having the other pixels fixed. After this process is completed, the result is smoothed.

After that the algorithm is repeated from the thresholding step (5.5) and several times iterated until convergence.

With this approach it is possible to get the images that contain only the grey levels known beforehand. The pixels that belong to the same grey level tend to lie together in the result of this approach. The rule that we change only the pixels that are situated on the border between different grey levels may be treated as implicit regularization.

This algorithm shows very good performance on the data with a small amount of projections. It is especially effective for binary images. It is capable of re-

constructing the object when other reconstruction algorithms fail to give reliable results.

As this algorithm is based on the other reconstruction algorithms, it is possible to take advantage of the both iterative algorithms together with knowledge about materials in the reconstructed object.

The performance of this approach on the complicated problem was demonstrated in the paper [2], where the approach was tested on the problem of reconstructing three-dimensional grains of complicated shape.

This approach was developed further in work [7], where authors improve the update step of the algorithm that improves the performance of the algorithm in some cases.

5.3 Simaultaneous Reconstruction and Segmentation Algorithm using the Hidden Markov Measure Field Model

The Simultaneous Reconstruction and Segmentation Algorithm that inspired our work was developed by M. Brady and Dominique Van de Sompel [10]. In this approach the reconstruction of the image is computed together with a segmentation and class parameters (mean values of the class and the standard deviations for each of the classes). We should underline that our work is based on this approach and, because of that, has many similarities with it.

The reconstruction is based on assumption that the image pixels belong to a set of classes, each with Gaussian distributed grey levels. The number of classes is assumed to be known a priori.

The segmentation in this approach is based on the Hidden Markov Measure Field Model (HMMFM) that we denote as δ.

In this approach the image $\mathbf{x}$ and the segmentation δ are computed as an opti-

mizers of the following energy function:

$$\begin{aligned}\mathbf{x}, \delta, \theta = \arg\max_{\mathbf{x},\delta,\theta} & \log p(\mathbf{b}|\mathbf{x}, \delta, \theta) + \\ & \log p(\mathbf{x}|\delta, \theta) + \\ & \log p(\delta) + \log p(\theta),\end{aligned} \tag{5.7}$$

where θ is the set of class parameters for classes, θ is a set of parameters of the classes. The parameters of classes are mean value μ_k and standard deviation σ_k. Thus, θ contains all the pairs (μ_k, σ_k). The logarithms of the terms have clear meaning and formulated as follows:

$$\log p(\mathbf{b}|\mathbf{x}, \delta, \theta) = \log p(\mathbf{b}|\mathbf{x}) = \sum_i b_i \log \sum_j a_{ij} x_j - \sum_j a_{ij} x_j \tag{5.8}$$

is a data fitting (or data fidelity term),

$$\log p(\mathbf{x}|\delta, \theta) = \sum_j \log \sum_k \delta_{jk} \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right) \tag{5.9}$$

is a class fitting term, and

$$\log p(\delta) = \lambda \sum_k \sum_j \sum_{j' \in N(j)} (\delta_{jk} - \delta_{j'k})^2 \tag{5.10}$$

is a regularization term for the HMMFM, where λ is a regularization parameter and

$$\log p(\theta) \tag{5.11}$$

is a regularization terms for the parameters of the classes (it may be different in case there is a knowledge about the class parameters and constant if there is no prior information about class parameters). In the paper [10] the optimization is done in three-steps fashion. First, the image $\mathbf{x}$ is computed having the HMMFM δ and parameters θ fixed. To do that the following problem is solved:

$$\arg\max_{\mathbf{x}} \log p(\mathbf{b}|\mathbf{x}) + \log \tilde{p}(\mathbf{x}|\delta, \theta), \tag{5.12}$$

where $\tilde{p}(\mathbf{x}|\delta, \theta)$ is an approximation of the term (8.14). The approximation is formulated in the following way:

$$\log \hat{p}(\mathbf{x}|\delta, \theta) = -\sum_j \frac{(x_j - \hat{\mu}_j)^2}{2\hat{\sigma}_j^2}, \tag{5.13}$$

where the parameters $\tilde{\mu}$ and $\tilde{\sigma}$ are computed according to the following rules:

$$\hat{\mu}_j = \sum_k \delta_{jk}\mu_k, \tag{5.14}$$

$$\hat{\sigma}_j^2 = \sum_k \delta_{jk}(\sigma_k^2 + \mu_k^2) - \hat{\mu}_j. \tag{5.15}$$

In the other words, the proability distribution $p(\mathbf{x}|\delta,\theta)$ in each pixel is approsimated with a single Gaussian.

After the image is updated, the HMMFM δ is computed having the image $\mathbf{x}$ and parameters θ fixed as an optimizer of the function

$$\arg\max_{\delta} \log p(\mathbf{x}|\delta,\theta) + \log p(\delta). \tag{5.16}$$

Finally, when the image and HMMFM are updated, the parameters of classes are reestimated having the image $\mathbf{x}$ and HMMFM δ fixed as follows:

$$\arg\max_{\theta} \log p(\mathbf{x}|\delta,\theta) + \log p(\theta). \tag{5.17}$$

We should note that we have tried to use this approach to compare our algorithm's results, but we have encountered an issue here: the problem (5.17) is non-convex and authors in the paper [10] are not specific about which approach they used for solving this problem.

5.4 Simultaneous Reconstruction and Segmentation without Neighborhood Prior

A similar problem was solved in [4], where the problem of joint reconstruction and segmentation was solved using similar Maximum a Posteriori Probability (MAP) aprroach, where the probability of the measured data was formulated as

$$\arg\max_{\mathbf{x},\delta,\theta} p(\mathbf{b}|\mathbf{x},\delta,\theta) = \arg\max_{\mathbf{x},\delta,\theta} p(\mathbf{b}|\mathbf{x})p(\mathbf{x}|\delta,\theta)p(\delta,\theta), \tag{5.18}$$

or, taking the logarithm of the objective function, one can get

$$\arg\max_{\mathbf{x},\delta,\theta} \log p(\mathbf{b}|\mathbf{x}) + \log p(\mathbf{x}|\delta,\theta) + \log p(\delta,\theta). \tag{5.19}$$

The data fitting term was formulated for the case where the data has a Poisson noise (for Transmission case):

$$\log p(\mathbf{b}|\mathbf{x}) = -\sum_i b_i \sum_j a_{ij} x_j - \sum_i \exp\left(-\sum_j a_{ij} x_j\right). \tag{5.20}$$

The grey values of the classes were treated as being distributed according to the Gamma distribution. Thus, the class fitting term $\log p(\mathbf{x}|\delta, \theta)$ is the logarithm of a mixture of Gamma distributions (θ are the parameters of these distributions).

$$\log p(\mathbf{x}|\delta, \theta) = \log \prod_j \sum_k \delta_{jk} p(x_j|k, \theta), \tag{5.21}$$

where $p(x_j|k, \theta)$ is a Gaussian distribution.

As the so-called hyperprior ($p(\delta, \theta)$) the following information is used: the mixture parameters (that we call HMMFM) are assumed to be distributed according to the Gamma distribution, the distribution parameters are also distributed as a Gamma distribution (for more details see the paper).

The optimization is done as an iterative two-steps algorithm. The first step is the image update with HMMFM δ and the parameters θ fixed:

$$\arg\min_{\mathbf{x}} \log p(\mathbf{b}|\mathbf{x}) + \log p(\mathbf{x}|\delta, \theta), \tag{5.22}$$

the second step is the HMMFM and parameters update. It is computed as a solution of the following optimization problem:

$$\arg\min_{\delta,\theta} \log p(\mathbf{x}|\delta, \theta) + \log p(\delta, \theta). \tag{5.23}$$

The update strategy for both steps may be found in the paper [4]. After both steps are complete, the next iteration is carried out starting with the image update.

5.5 Other Related Work

Similar ideas of doing the reconstruction and segmentation simultaneously were used in the paper [8]. This paper deals with the problem of ultrasound imaging which uses classes as priors for denoising the resulting image.

In the work [11] the authors deal with the problem of reconstructing the image from noisy data (the cases of tomographic reconstructions and image deblurring were considered) with l_0 regularization . The problem is formulated as follows:

$$\arg\min_{\mathbf{x}} \|A\mathbf{x} - \mathbf{b}\|_2^2 + \lambda I(\|D_j\mathbf{x}\|), \tag{5.24}$$

where I is an indicator function that is equal to one when the argument is non-zero and is equal to zero, when the argument is zero. As was already mentioned, this objective function penalizes the borders between different grey level areas of the image. As a result, the larger the regularization parameter λ the less borders are presented in the result. Thus, this regularization produces piecewise constant reconstructions, where regions where the colour is constant may be treated as the pixels that belong to the same class.

As was already mentioned, this regularization term is non-convex. Because of it that, this way of dealing with this problem is quite inconvenient to solve as it is NP-hard in general. In this work authors compute an approximation to the solution of this problem using the fact that in case $\mathbf{x}$ is 1D array and the matrix A is an identity matrix, this problem becomes not NP-hard.

In the work [9] the authors deal with problem of reconstruction and segmentation by means of a level set approach for the CT problems. The problem that is being solved is formulated as follows:

$$\arg\min_{\mathbf{x}} \|A\mathbf{x} - \mathbf{b}\|_2^2 + \lambda|\Gamma|, \tag{5.25}$$

where Γ is a set of points, in which the image $\mathbf{x}$ may have discontinuities and λ is a regularization parameter. In other words, the problem that is solved is similar to reconstruction with l_0 regularization.

The minimization algorithm runs as follows:

1. The initial approximation for the set of points Γ_0, where the image is allowed to have singularities is selected.
2. After that the image $\mathbf{x}$ is optimized varying the pixels that belong to the set Γ and the grey levels of the other regions that are separated by the set Γ.
3. After this the set of points with discontinuities is changed using the Level-Sets approach.
4. Repeat steps 2 and 3 until convergence.

For the technical details of the update steps please refer to the paper.

This approach was further developed in the paper [6] where the authors apply this approach to the SPECT/CT problem. A similar approach was presented in the work [5].

Another interesting approach to the reconstruction and segmentation is discussed in the paper [12]. In this work the object jth element x_j of the solution $\mathbf{x}$ is represented in the form

$$x_j = x_{1j}H(\phi(j)) + x_{2j}(1 - H(\phi(j))), \tag{5.26}$$

where H is a Heaviside function that is equal to 1 when the argument is greater than zero, and is equal to zero otherwise, the vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ are the smooth images that describe the object in different adjacent regions and the function $\phi(j)$ is a level set function that separates the regions.

The problem is formulated as follows:

$$\begin{aligned} \arg\min_{\mathbf{x}_1,\mathbf{x}_2,\phi} \|A\mathbf{x} - \mathbf{b}\|_2^2 + \lambda_1 \sum_j \|D_j\mathbf{x}_1\|_2^2 H(\phi(j)) + \\ \lambda_2 \sum_j \|D_j\mathbf{x}_2\|_2^2 H(\phi(j)) + \lambda_3 \sum_j \|D_j H(\phi(j))\| \\ \text{s.t.} \quad \forall j \quad l_1 \leq x_{1j} \leq h_1, l_2 \leq x_{2j} \leq h_2 \end{aligned} \tag{5.27}$$

where λ_1, λ_2, λ_3 are the regularization parameters, D_j is a discrete approximation of the gradient, l_1 and l_2 are the lowest allowed values for $\mathbf{x}_1$ and $\mathbf{x}_2$ respectively, h_1 and h_2 are the highest values of $\mathbf{x}_1$ and $\mathbf{x}_2$.

Thus, the reconstruction here is done in the form of regularized optimization problem, the segmentation is done using the Level Sets approach.

The optimization procedure of this problem is an iterative process with two steps. First, the images $\mathbf{x}_1$ and $\mathbf{x}_2$ are updated having the function $\phi(j)$ fixed. Then the function $\phi(j)$ is updated having the images $\mathbf{x}_1$ and $\mathbf{x}_2$ fixed. After the function ϕ is updated, the images are recomputed again and so on until convergence.

The optimization of the images is carried out using the Gradient Projection Congugate Gradient approach (GPCG).

For the details about the reconstruction process and for results please see the paper.

Using this procedure it is possible to get the objects that have some areas with smooth intensity levels inside of the larger areas with smooth intensity areas (though it is important that three different areas do not touch each other as it is not possible to represent three different regions that touch each other with only two component images as shown in (5.26)). The object have sharp edges between different smooth areas.

5.6 Place of Our Work in This Context

In our work we want to use additional knowledge about the reconstructed object in order to get better reconstruction. We selected the Hidden Markov Measure Field Model (HMMFM) approach to be our segmentation model for that reasons because it is real-valued (i.e., non-discrete), probabilistic model. It is easy to regularize and it is not hard to solve the optimization problem as the standard optimization techniques are applicable here.

It is possible to make link between our HMMFM-based approach and DART: the first step of DART iteration (i.e. the thresholding) may be treated as an approximation of the HMMFM segmentation, the algebraic (SIRT) step may be treated as computation of next image approximation. The last is valid because the l_2 regularized HMMFM will have values different from 1 and 0 for some of the classes on the borders of the area with the same grey level inside, and because of that the grey level is more likely to change near the edge of the area than inside of the area.

For our HMMFM we need the following data: the matrix A, the data $\mathbf{b}$, the mean values μ_k and standard deviations σ_k for each class and two regularization parameters λ_1 and λ_2. The set of the input parameters is quite similar to what is needed for a DART algorithm.

Bibliography

[1] Kees Joost Batenburg and Jan Sijbers. "DART: a fast heuristic algebraic reconstruction algorithm for discrete tomography". In: *Image Processing, 2007. ICIP 2007. IEEE International Conference on.* Vol. 4. IEEE. 2007, pp. IV–133.

[2] KJ Batenburg et al. "DART: a robust algorithm for fast reconstruction of three-dimensional grain maps". In: *Journal of Applied Crystallography* 43.6 (2010), pp. 1464–1473.

[3] Maarten Depypere et al. "The minimal entropy prior for simultaneous reconstruction and segmentation of in vivo microCT trabecular bone images". In: *Biomedical Imaging: From Nano to Macro, 2009. ISBI'09. IEEE International Symposium on.* IEEE. 2009, pp. 586–589.

[4] Ing-Tsung Hsiao, Anand Rangarajan, and Gene Gindi. "Joint-MAP reconstruction/segmentation for transmission tomography using mixture-models as priors". In: *Nuclear Science Symposium, 1998. Conference Record. 1998 IEEE.* Vol. 3. IEEE. 1998, pp. 1689–1693.

[5] Esther Klann. "A Mumford-Shah-like method for limited data tomography with an application to electron tomography". In: *SIAM Journal on Imaging Sciences* 4.4 (2011), pp. 1029–1048.

[6] Esther Klann, Ronny Ramlau, and Wolfgang Ring. "A Mumford-Shah level-set approach for the inversion and segmentation of SPECT/CT data". In: *Inverse Probl. Imaging* 5.1 (2011), pp. 137–166.

[7] F Javier Maestre-Deusto et al. "ADART: An adaptive algebraic reconstruction algorithm for discrete tomography". In: *Image Processing, IEEE Transactions on* 20.8 (2011), pp. 2146–2152.

[8] Marcelo Pereyra et al. "Segmentation of skin lesions in 2-D and 3-D ultrasound images using a spatially coherent generalized Rayleigh mixture model". In: *Medical Imaging, IEEE Transactions on* 31.8 (2012), pp. 1509–1520.

[9] Ronny Ramlau and Wolfgang Ring. "A Mumford–Shah level-set approach for the inversion and segmentation of X-ray tomography data". In: *Journal of Computational Physics* 221.2 (2007), pp. 539–557.

[10] Dominique Van de Sompel and Michael Brady. "Simultaneous reconstruction and segmentation algorithm for positron emission tomography and transmission tomography." In: *ISBI*. 2008, pp. 1035–1038.

[11] Martin Storath et al. "Joint image reconstruction and segmentation using the Potts model". In: *Inverse Problems* 31.2 (2015), p. 025003.

[12] Sungwon Yoon, Angel R Pineda, and Rebecca Fahrig. "Simultaneous segmentation and reconstruction: A level set method approach for limited view computed tomography". In: *Medical physics* 37.5 (2010), pp. 2329–2340.

Chapter 6

Simultaneous Tomographic Reconstruction and Segmentation with Class Priors

We consider tomographic imaging problems where the goal is to obtain both a reconstructed image and a corresponding segmentation. A classical approach is to first reconstruct and then segment the image; more recent approaches use a discrete tomography approach where reconstruction and segmentation are combined to produce a reconstruction that is identical to the segmentation. We consider instead a hybrid approach that simultaneously produces both a reconstructed image and a segmentation. We incorporate priors about the desired classes of the segmentation through a Hidden Markov Measure Field Model, and we impose a regularization term for the spatial variation of the classes across neighboring pixels. We also present an efficient implementation of our algorithm based on state-of-the-art numerical optimization algorithms. Simulation experiments with artificial and real data demonstrate that our combined approach can produce better results than the classical two-step approach.

6.1 Introduction

In computed tomography (CT) it is often the case that the reconstructed 2D or 3D image is also used as an intermediate result in order to arrive at a segmentation of the reconstructed object. The purpose of the segmentation is, e.g., to separate an object from the background [36] or to identify specific objects or regions [4], and this has many applications in medical imaging and in non-destructive testing in materials science. The classical approach is first to produce a 2D or 3D reconstruction, followed by segmentation of this image. Both steps are likely to introduce errors and artifacts, and the errors in the reconstructed image usually propagate to the segmentation.

In order to avoid this one can try to combine the image reconstruction and segmentation in a single reconstruction model (leading to a more complex problem). One idea is to allow only a small number of (known or unknown) pixel values, leading to the problem of discrete tomography – see [4], [3], [14] for examples. Another idea is to use segmentation principles to identify objects with known pixel values in order to obtain sharper edges around these objects [36]. In both cases, the segmented image is identical to the reconstructed image.

In this work we take a different approach, which we refer to as *Simultaneous Reconstruction and Segmentation* (SRS), where we simultaneously produce both a reconstructed image and a segmentation. This idea was originally proposed by Van de Sompel and Brady [33] who used a Hidden Markov Measure Field Model (HMMFM) [24] that "learns" information about the different types of objects or phases from the given data during the SRS process. They demonstrate that the combined approach is able to produce good reconstructions with sharp boundaries of the objects. A different algorithm that simultaneously reconstructs and segments directly from the CT data was proposed by Ramlau and Ring [30], and later generalized to SPECT in [22]. The same approach was also applied to CT problems with limited data in [21]. Other papers that describe the use of a hidden Markov field are [25] and [35].

Instead of "learning" or estimating information about the segmentation classes during the reconstruction phase, we want to use this information as a prior to stabilize the solution. We therefore propose a variant of the above method where we explicitly specify prior information about the different classes (i.e., about different types of phases), to be used in the segmentation via the HMMFM.

There is a variety of computational algorithms for tomographic reconstruction

that use regularization in order to deal with the difficulties of the underlying inverse problem [16]. Filtered back projection and its extension to 3D geometries are very popular algorithms [11], [19] that work well when enough data/projections are available and the noise is not too large. Algebraic iterative reconstruction methods [17] are also used frequently, and they can give better reconstructions in case of limited data. Yet other algorithms are based on variational formulations where one minimizes a combination of a data-fitting term and a regularization term (such as [6], [32]) that penalizes unwanted features; these methods can be more flexible and compensate for large noise and limited data. While TV regularization has been very popular in recent years, it still has some limitations such as smearing of texture-rich regions and staircasing [34].

Concerning segmentation techniques, there are many approaches to the problem of extracting regions of the image, e.g., by modeling the boundary between regions either using implicit contour models like the level-set methods [31], [28], [1] or including an explicit boundary model as in snakes [20]. Another group of methods are based on pixel labeling, e.g., modeled by a Markov Random Field where solutions can be obtained efficiently using graph-cut methods [10], [8]. In order to segment texture-rich images methods based on texture-features like the structure tensor [12] are popular, but also sparse methods have shown to be effective for texture segmentation [23], [15]. Some methods however require manual input [20], [15], [26] in order to work correctly. In CT all of the methods may be used, but it is preferable to use methods that do not need initialization by the user.

The classical approach with two separate steps (first reconstruct the image and then segment it) has proven to be quite effective. However, although prior knowledge about the expected types of objects or phases can be used during the segmentation, this knowledge is rarely used in the reconstruction step. Discrete tomography provides one way to incorporate such information about the desired pixel intensities.

In this paper we consider a general SRS framework and we show how various assumptions about the problem lead to a specific computational problem that can be handled by means of state-of-the-art numerical optimization methods. We then use numerical simulations to show that our algorithm can give very good segmentation result for problems where the parameters of the reconstructed phases are known, because our specific use of this prior information shrinks the range of possible solutions and hence increases the quality of the reconstructed image.

Our algorithm employs a user-specified HMMFM to assign to each pixel the prob-

abilities for belonging to the given classes. In this way we give a quantitative measure of the similarity between each reconstructed pixel and each given class. This approach to the segmentation problem allows us to naturally use the connection between the HMMFM and the posterior probability density distribution. Our computational algorithm uses efficient numerical optimization algorithms, thus avoiding the slowly converging sampling methods.

Recall that X-ray CT is based on the principle that if we send X-rays through an object and measure the damping of each ray then, with infinitely many rays, we can perfectly reconstruct the object. The attenuation of an X-ray is proportional to the object's attenuation coefficient, as described by Lambert-Beer's law [13, §2.3.1]. We divide the domain onto pixels whose unknown nonnegative attenuation coefficients are organized in the vector $\boldsymbol{x} \in \mathbb{R}^N$. Similarly we organize the measured damping of the rays into the vector $\boldsymbol{b} \in \mathbb{R}^M$. Then we obtain a linear system of equations $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$ with a large sparse *system matrix* governed solely by the geometry of the measurements: element a_{ij} is the length of the ith ray passing through pixel j, and the matrix is sparse because each ray only hits a small number of pixels [27].

Our paper is organized as follows. In Section 6.2 we present our general formulation of the reconstruction model, and in Section 6.3 we make our model more specific by means of assumptions related to the CT problem. Section 6.4 presents our considerations related to an efficient computational algorithm for the SRS problem. Our numerical experiments are presented in Section 6.5, and we finish the paper with a short conclusion. Throughout the paper we use the following notation:

- i is the data index.
- j is the pixel index.
- k is the class index.
- $\boldsymbol{b}$ is a vector with the measured data $b_1, b_2, \ldots, b_M$.
- $\boldsymbol{x}$ is a vector with the pixels $x_1, x_2, \ldots, x_N$ of the image (the columns of the image are stacked into one long vector); the pixel values are the attenuation coefficients of the object.
- $\boldsymbol{A}$ is the $M \times N$ system matrix for the CT problem which describes the linear relationship $\boldsymbol{b} = \boldsymbol{A}\,\boldsymbol{x}$ between the attenuation coefficients and the data.

- $\delta = \{\delta_{jk}\}$ is the set of probabilities in the HMMFM for each class k of the object and for each pixel j, satisfying $\sum_{k=1}^{K} \delta_{jk} = 1$ for all j.
- $p(\boldsymbol{x}, \delta \,|\, \boldsymbol{b})$ is the posterior probability density function for the image and the HMMFM, given the data.

In each pixel, the probabilities for all the classes of the HMMFM must sum to one. We note that the above notation trivially generalizes to 3D problems where x_j denotes a voxel instead of a pixel.

6.2 Problem Formulation

To solve inverse problems it is necessary to incorporate prior information about the solution, in order to obtain a well-defined problem and a stable solution. In this work, our prior takes the form of explicit information about the different types of phases in the object, which provides a rigorous prior for the segmentation. Specifically, we assume that the object's attenuation coefficients (the pixels of the reconstruction) belong to K classes, where K is much smaller than the number N of pixels. We assume that K is known and we provide prior information about these classes in the form of probability densities associated with the classes. The parameters of the different classes (the mean attenuation coefficients μ_k and their variances σ_k for $k = 1, \ldots, K$) are assumed to be known from previous studies; see §6.3.2. Moreover, following [24] we use an HMMFM to incorporate a spatial prior inspired by the framework introduced in [33].

Our reconstruction problem originates from a classical constrained Bayesian formulation:

$$\begin{aligned} \underset{\boldsymbol{x},\delta}{\text{argmax}} \quad & p(\boldsymbol{x}, \delta \,|\, \boldsymbol{b}) \\ \text{s.t.} \quad & \sum_{k=1}^{K} \delta_{jk} = 1, \;\; \delta_{jk} \geq 0, \quad j = 1, \ldots, N, \;\; k = 1, \ldots, K. \end{aligned} \tag{6.1}$$

In accordance with Bayes' rule, the probability $p(\boldsymbol{x}, \delta \,|\, \boldsymbol{b})$ in (6.1) can be formulated as:

$$p(\boldsymbol{x}, \delta \,|\, \boldsymbol{b}) = \frac{p(\boldsymbol{b} \,|\, \boldsymbol{x}, \delta)\, p(\boldsymbol{x} \,|\, \delta)\, p(\delta)}{p(\boldsymbol{b})}, \tag{6.2}$$

where

- $p(\boldsymbol{b} \mid \boldsymbol{x}, \delta) = p(\boldsymbol{b} \mid \boldsymbol{x})$ is the probability of obtaining the data $\boldsymbol{b}$ given the image $\boldsymbol{x}$; the data does not depend on the segmentation of the image.
- $p(\boldsymbol{x} \mid \delta)$ is the probability of $\boldsymbol{x}$ given the probabilities of each class in each pixel.
- $p(\delta)$ expresses our belief in the HMMFM; usually, the more complex or chaotic the δ, the less we trust in it.
- $p(\boldsymbol{b})$ is a normalization constant.

The function given in (6.2) is a product of several probabilities, and to make the optimization process easier (as is common) we consider the logarithm of the objective function instead of the function itself. Thus, ignoring the normalization constant $p(\boldsymbol{b})$, we arrive at the general formulation of the SRS problem:

$$\begin{aligned}(\boldsymbol{x}^*, \delta^*) &= \underset{\boldsymbol{x},\delta}{\operatorname{argmax}} \log p(\boldsymbol{x}, \delta \mid \boldsymbol{b}) \\ &= \underset{\boldsymbol{x},\delta}{\operatorname{argmax}} \Big(\log p(\boldsymbol{b} \mid \boldsymbol{x}) + \log p(\boldsymbol{x} \mid \delta) + \log p(\delta) \Big) \qquad (6.3) \\ \text{s.t.} \quad & \sum_{k=1}^{K} \delta_{jk} = 1, \ \delta_{jk} \geq 0, \quad j = 1, \ldots, N, \ k = 1, \ldots, K.\end{aligned}$$

This model can be used for simple classification situations (e.g., graylevel/intensity classification) as well as for more complex classification algorithms. In this paper we will use and analyze this framework for a simple graylevel problem.

Having solved the problem for $\boldsymbol{x}$ and δ, it is easy to obtain the segmentation knowing the HMMFM. For each pixel x_j we compute the corresponding labels s_j as:

$$s_j \equiv \underset{k}{\operatorname{argmax}} \ \delta_{jk}, \qquad j = 1, \ldots, N \qquad (6.4)$$

which is simply the most probable class in each pixel. Hereby we both obtain a reconstructed image and a segmentation based on specific model assumptions.

6.3 Assumptions for the CT Reconstruction Problem

To make the general SRS framework (6.3) more specific we make assumptions about the probability density functions that we use. Here we consider them as

known, and indeed in many cases we will know the attenuation coefficients and their uncertainties for the phases under study (e.g., muscle, fat, bone, and air in medical imaging). How to obtain these parameters is not part of this work.

6.3.1 Data Fitting Term

Our first assumption regards the noise model. According to [2], the measured data is usually a sum of several terms:

- Data received from the X-ray illumination of the object, with Poisson noise.
- Poisson noise of the measuring equipment and from external sources.
- Gaussian noise caused by the electronics and the conversion from an analog signal to digital data.

Following [2] we assume that the signal-to-noise level is high enough that we can approximate the total noise by additive unbiased Gaussian noise $\mathcal{N}(0, \sigma^2_{\text{noise}})$ where σ_{noise} is the standard deviation of the noise. We also assume that the noise is uncorrelated with the data and independent such that $p(\boldsymbol{b} \,|\, \boldsymbol{x}) = p(\boldsymbol{\varepsilon})$, where the vector $\boldsymbol{\varepsilon}$ represents the Gaussian noise, i.e., $\varepsilon_i \sim \mathcal{N}(0, \sigma^2_{\text{noise}})$. Consequently the first term in our reconstruction model, which we refer to as the *data fidelity term*, is given by:

$$\begin{aligned} p(\boldsymbol{b} \,|\, \boldsymbol{x}) &= \prod_{i=1}^{M} p(b_i \,|\, \boldsymbol{x}) = \frac{1}{\sqrt{2^M \pi^M} \sigma^M_{\text{noise}}} \exp\left(- \sum_{i=1}^{M} \frac{(\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b})_i^2}{2\sigma^2_{\text{noise}}} \right) \\ &= \frac{1}{\sqrt{2^M \pi^M} \sigma^M_{\text{noise}}} \exp\left(- \frac{\|\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b}\|_2^2}{2\sigma^2_{\text{noise}}} \right), \end{aligned}$$

where $(\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b})_i$ denotes the ith element of the residual vector. The first term in (6.3) thus becomes:

$$\log p(\boldsymbol{b} \,|\, \boldsymbol{x}) = - \frac{\|\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b}\|_2^2}{2\sigma^2_{\text{noise}}} - \frac{1}{2} \log(2^M \pi^M \sigma^{2M}_{\text{noise}}), \tag{6.5}$$

and we note that $\frac{1}{2} \log(2^n \pi^n \sigma^{2n}_{\text{noise}})$ is a constant that is unaffected by the optimization. The data fidelity term in this problem formulation is identical to a classical least squares fitting term.

6.3.2 Class Fitting Term

To specify the second term of (6.3) related to the classes of the image, we assume that the object is composed of a set of K different phases, and that each phase has the same attenuation coefficient everywhere. Specifically, we assume a Gaussian mixture where we specify the distribution of the attenuation coefficients within a class as a normal distribution with mean value μ_k equal to the expected attenuation coefficient and a (small) standard deviation σ_k. Here, all μ_k and σ_k for $k = 1, \ldots, K$ are assumed to be known, and we have

$$p(x_j \,|\, \text{class} = k) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right).$$

Now let $\delta_j = \{\delta_{j1}, \ldots, \delta_{jK}\}$ denote the set of the K class probabilities associated with pixel j. Then we can write the probability $p(\boldsymbol{x} \,|\, \delta)$ as

$$p(\boldsymbol{x} \,|\, \delta) = \prod_{j=1}^{N} p(x_j \,|\, \delta_j) = \prod_{j=1}^{N} \sum_{k=1}^{K} p(x_j \,|\, \text{class} = k)\, p(\text{class} = k \,|\, \delta_j).$$

Since we assume that $p(\text{class} = k \,|\, \delta_j) = \delta_{jk}$ is the probability of the pixel j belonging to class k, the above expression takes the form:

$$p(\boldsymbol{x} \,|\, \delta) = \prod_{j=1}^{N} \sum_{k=1}^{K} \delta_{jk} \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right).$$

Thus, the second term of (6.3) can be written as

$$\log p(\boldsymbol{x} \,|\, \delta) = \sum_{j=1}^{N} \log\left[\sum_{k=1}^{K} \frac{\delta_{jk}}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right)\right]. \tag{6.6}$$

We refer to this as the *class-fitting term.*

6.3.3 Regularization Term

The third term of (6.3) is our confidence in the segmentation and it is thus a *regularization term* where we specify our prior knowledge about the behavior of the segmentation. Contrary to classical regularization methods, our prior is

concerned with the classes (and not the pixel values). We use the following generic formulation

$$\log p(\delta) = -\sum_{k=1}^{K} R(\delta_k), \tag{6.7}$$

where $\delta_k = \{\delta_{1k}, \ldots, \delta_{Nk}\}$ is the set of probabilities for class k and R is a function that expresses our prior, such as our knowledge of the expected spatial correlations of the classes among the pixels. In this paper, we use two different functions inspired by common choices in image reconstruction, cf. [16].

For ease of presentation, consider the case where $\boldsymbol{x}$ represents a 2D image (the extension to 3D is obvious) and let j' and j'' denote indices to the two elements of $\boldsymbol{x}$, in its 2D representation, that are neighbors horizontally and vertically. Let $\mathcal{J}$ denote the set of indices j for which both j' and j'' refer to pixels inside the image.

- The total variation (TV), i.e., the 1-norm of the gradient magnitude for the kth class associated with all pixels of $\boldsymbol{x}$:

$$R_{\mathrm{TV}}(\delta_k) = \sum_{j \in \mathcal{J}} \left((\delta_{jk} - \delta_{j'k})^2 + (\delta_{jk} - \delta_{j''k})^2 \right)^{1/2}. \tag{6.8}$$

- A Tikhonov-type regularizer equal to the squared 2-norm of the gradient magnitude for the kth class associated with all pixels of $\boldsymbol{x}$:

$$R_{\mathrm{Tik}}(\delta_k) = \sum_{j \in \mathcal{J}} \left((\delta_{jk} - \delta_{j'k})^2 + (\delta_{jk} - \delta_{j''k})^2 \right). \tag{6.9}$$

The use of the TV function R_{TV} (6.8) allows discontinuities in the probabilities for the classes associated with neighboring pixels – and one expects this to be well suited for the segmentation process. The use of the Tikhonov function R_{Tik} (6.9) enforces some spatial smoothness of the probabilities among classes associated with neighboring pixels.

6.3.4 The Reconstruction Model

At this time we assemble the three terms specified above into the complete reconstruction model. In doing so, we introduce two *regularization parameters* λ_{noise}

and λ_{class} that let us balance the weight given to the noise term and the regularization term, respectively, and as is common we absorb the factor $1/(2\sigma^2_{\text{noise}})$ into λ_{noise}. The complete reconstruction model thus takes the following form (swapping "min" for "max" and changing signs of the terms):

$$\begin{aligned}(\boldsymbol{x}^*, \delta^*) = \underset{\boldsymbol{x},\delta}{\operatorname{argmin}} \Bigg(& \lambda_{\text{noise}} \|\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \lambda_{\text{class}} \sum_{k=1}^{K} R(\delta_k) \\ & - \sum_{j=1}^{N} \log \left[\sum_{k=1}^{K} \frac{\delta_{jk}}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right) \right] \Bigg) \\ \text{s.t.} \quad & \sum_{k=1}^{K} \delta_{jk} = 1,\ \delta_{jk} \geq 0, \quad j = 1, \ldots, N,\ k = 1, \ldots, K.\end{aligned} \tag{6.10}$$

To summarize, our reconstruction problem (6.10) takes as input our data $\boldsymbol{b}$ and the class parameters μ_k, σ_k, $k = 1, \ldots, K$, together with the regularization function $R(\cdot)$ and the regularization parameters λ_{noise}, λ_{class}, and produces the solution pair $\boldsymbol{x}^*$ and δ^*. We note that the reconstruction problem is non-convex in $\boldsymbol{x}$ and our computational algorithm (described in the next section) must take this into account.

6.3.5 Simplifications

To deal with the non-convexity of the problem, we introduce two different simple convex approximations to the class-fitting term $p(\boldsymbol{x} \mid \delta)$, and both of them are used in the algorithm described in the next section. Note that the objective function in (6.10) is a multi-modal function for each pixel, consisting of a sum of Gaussian functions, that causes the non-convexity of the problem. The underlying idea is then to approximate this function with a uni-modal approximation – a single Gaussian function – such that we can use standard methods from convex optimization.

The first type of simplification was introduced in [33], and for each pixel x_j it "lumps" the parameters δ_{jk}, μ_k and σ_k into the parameters $\hat{\mu}_j$, $\hat{\sigma}_j$ and $\hat{\delta}_j = 1$, where

$$\hat{\mu}_j = \sum_{k=1}^{K} \delta_{jk}\mu_k, \quad \hat{\sigma}_j^2 = \sum_{k=1}^{K} \delta_{jk}(\sigma_k^2 + \mu_k^2) - \hat{\mu}_j^2, \qquad j = 1, \ldots, N. \tag{6.11}$$

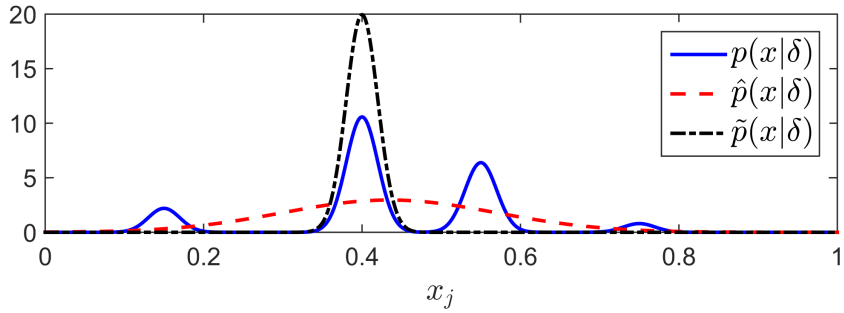

Figure 6.1: Illustration of the simplified uni-modal class-fitting functions $\hat{p}(\boldsymbol{x} \,|\, \delta)$ (6.12) and $\tilde{p}(\boldsymbol{x} \,|\, \delta)$ (6.13) together with the multi-modal function $p(\boldsymbol{x} \,|\, \delta)$ (6.6). We use the parameters $K = 4$, $\delta_j = \{0.11, 0.53, 0.32, 0.04\}$, $\mu_1 = 0.15$, $\mu_2 = 0.4$, $\mu_3 = 0.55$, $\mu_4 = 0.75$, and $\sigma_1 = \sigma_2 = \sigma_3 = \sigma = 4 = 0.02$.

The simplified function then takes the form (omitting an additive constant):

$$\log \hat{p}(\boldsymbol{x} \,|\, \delta) = -\sum_{j=1}^{N} \frac{(x_j - \hat{\mu}_j)^2}{2\hat{\sigma}_j^2}. \tag{6.12}$$

The second type of simplification takes a different approach. When we are close to the solution we expect that for most of the pixels x_j the corresponding δ_{jk} will approach a situation where one of them, say, δ_{jk_j}, is close to 1 while the remaining ones are small – corresponding to a high probability that this pixel belongs to class k_j. In this situation it is natural to set

$$\hat{\delta}_{jk} = \begin{cases} 1, \text{ if } k = k_j, \\ 0, \text{ otherwise,} \end{cases} \qquad j = 1, \ldots, N$$

and then (6.11) simplifies to $\hat{\mu}_j = \mu_{k_j}$, $\hat{\sigma}_j = \sigma_{k_j}$, and we obtain the simplified function

$$\log \tilde{p}(\boldsymbol{x} \,|\, \delta) = -\sum_{j=1}^{N} \frac{(x_j - \mu_{k_j})^2}{2\sigma_{k_j}^2}. \tag{6.13}$$

Figure 6.1 illustrates these approximations.

6.4 Algorithm

We compute a solution to the reconstruction problem (6.10) using an iterative, alternating optimization algorithm as is common practice for non-convex problems. Our algorithm is based on the following idea. First we use the simplification in (6.12) to roughly approximate the solution. When we are close to the solution, we instead use the simplification (6.13) in order to improve the reconstruction result. Hence the algorithm has two overall stages, and in both stages the iterations alternate between two steps in which we update either the image or the classes. In the first stage we compute an approximate solution pair $(\hat{\boldsymbol{x}}^*, \hat{\delta}^*)$ using the simplified expression (6.12). This approximation is then used as initial guess for the second stage based on the alternative simplified function in (6.13). The algorithm is summarized in Fig. 6.2 at the end of this section.

6.4.1 Stage 1

In the first stage we apply our iterative scheme to problem (6.10) with the simplifications (6.11) and (6.12). Our algorithm alternates between two steps in which we update either $\hat{\boldsymbol{x}}^n$ or $\hat{\delta}^n$, where the superscript n denotes the iteration. In the *image-update step* we update the pixel values:

$$\hat{\boldsymbol{x}}^{n+1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \left(\lambda_{\text{noise}} \|\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \sum_{j=1}^{N} \frac{(x_j - \hat{\mu}_j^n)^2}{2(\hat{\sigma}_j^n)^2} \right), \tag{6.14}$$

and we note that $\hat{\mu}_j^n$ and $\hat{\sigma}_j^n$ depend on the iteration – they are functions of the current HMMFM and computed using (6.11)). The function in (6.14) is convex in $\boldsymbol{x}$ and we compute its minimum by means of the Conjugate Gradient Least Squares (CGLS) algorithm [7] which is well suited for large-scale problems. As the initial guess for CGLS we use the image from the previous iterate.

In the *class-update step* we update the classes:

$$\hat{\delta}^{n+1} = \underset{\delta}{\operatorname{argmin}} \left(\lambda_{\text{class}} \sum_{k=1}^{K} R(\delta_k) - \sum_{j=1}^{N} \log \left[\sum_{k=1}^{K} \frac{\delta_{kj}}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(\hat{x}_j^n - \mu_k)^2}{2\sigma_k^2} \right) \right] \right) \tag{6.15}$$

$$\text{s.t.} \quad \sum_{k=1}^{K} \delta_{jk} = 1, \quad \delta_{jk} \geq 0, \qquad j = 1, \ldots, N, \quad k = 1, \ldots, K.$$

Different algorithms can be used to solve this problem; we use the iterative Frank-Wolfe algorithm [5], also known as the conditional gradient method, which enforces the conditions (for $j = 1, \ldots, N$ and $k = 1, \ldots, K$)

$$\sum_{k=1}^{K} \hat{\delta}_{jk}^{n+1} = 1 \qquad \text{and} \qquad \hat{\delta}_{jk}^{n+1} > 0.$$

We choose the initial guess $\hat{\delta}$ for this algorithm as $\hat{\delta}_{jk}^{0} = 1/K$, for all j and k, which by numerical experiments was found to be the most robust initialization.

6.4.2 Stage 2

When the changes in the image $\hat{\boldsymbol{x}}^n$ and the classes $\hat{\delta}^n$ become small during Stage 1, this signals that we are making little progress with the approach based on the simplifications (6.11) and (6.12). We stop these iterations when

$$\|\hat{\boldsymbol{x}}^{n+1} - \hat{\boldsymbol{x}}^n\|_2 / \|\hat{\boldsymbol{x}}^n\|_2 \leq 10^{-6}. \tag{6.16}$$

At this stage, most of the pixels j in the image have been assigned to a single class, meaning that the corresponding δ_{jk} are close to 0 or 1 and $\hat{\sigma}_j \ll 1$.

It is now natural to switch to Stage 2 in which we use the second type of simplification in (6.13) where, for each pixel, we only use the class with the highest probability. The main goal of this stage is to consider those remaining pixels for which δ_{jk} have not yet approached 0 or 1, which typically are those at the edges inside the image. Equation (6.13) ensures that these pixels are also assigned to a single class, forcing $\hat{\sigma}_j \ll 1$. We emphasize that, according to our experiments, Stage 2 will give good results only when most pixels have been assigned to a single class in Stage 1.

First we perform the image-updating step, which now takes the form:

$$\boldsymbol{x}^{n+1} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \left(\lambda_{\text{noise}} \|\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \sum_{j=1}^{N} \frac{(x_j - \mu_{k_j})^2}{2\sigma_{k_j}^2} \right), \tag{6.17}$$

where k_j is the index to the largest δ_{jk} for the jth pixel. We still use CGLS to solve (6.17). The class-updating step is identical to that of Stage 1, and again we solve it with Frank-Wolfe algorithm. Usually it is enough to perform 1–3 iterations of the Stage 2. Our observation is that the fine-tuning of the classification results in an improved reconstruction.

Initialization: $\hat{\delta}^0_{jk} = 1/K$, $\forall j, k$.

Stage 1

For $n = 1, 2, \ldots, n_1$

Compute $\hat{\mu}_j$ and $\hat{\sigma}_j$ using (6.11).

Compute $\hat{\boldsymbol{x}}^n$ using (6.14) by means of CGLS with initial guess $x_j = \hat{\mu}_j^{n-1}$.

Compute $\hat{\delta}^n$ using (6.15) by means of FWA with initial guess $\hat{\delta}^{n-1}$.

Stage 2

Set $\boldsymbol{x}^{n_1} = \hat{\boldsymbol{x}}^{n_1}$ and $\delta^{n_1} = \hat{\delta}^{n_1}$.

For $n = n_1 + 1, n_1 + 2, \ldots, n_1 + n_2$

Compute k_j such that $\delta_{jk_j} = \max_k \delta_{jk}$, $\forall j$.

Compute $\boldsymbol{x}^n$ using (6.17) by means of CGLS with initial guess $x_j = \mu_{k_j}^{n-1}$.

Compute δ^n using (6.15) by means of FWA with initial guess δ^{n-1}.

Finish: Set $\boldsymbol{x}^* = \boldsymbol{x}^{n_1+n_2}$ and $\delta^* = \delta^{n_1+n_2}$.

Figure 6.2: Summary of our SRS algorithm; "CGLS" is a robust implementation of the Conjugate Gradient algorithm for Least Squares problems, and "FWA" denotes the Frank-Wolfe algorithm. Note that an initial image is not needed for the SRS algorithm.

6.5 Numerical Results

In this section we present a series of numerical experiments where we compare our two-stage algorithm SRS with the classical approach where the reconstruction and segmentation are performed in two consecutive independent steps. We consider two versions of our algorithm, SRS-TV and SRS-Tik, corresponding to the two different regularization terms (6.8) and (6.9), respectively. We use quite small test images such that the artifacts of the different algorithms are clearly visible.

For the reconstruction step of the classical approach we use two different algorithms: the Filtered Back Projection (FBP) method as implemented in MATLAB's `iradon` function, and Total Variation (TV) regularization as implemented in the software package[1] TVReg [18] which solves the problem

$$\min_{\boldsymbol{x}} 1/2 \|\boldsymbol{A}\,\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \alpha\,\mathrm{TV}(\boldsymbol{x}), \qquad 0 \le x_i \le 1, \quad i = 1, \ldots, N, \tag{6.18}$$

[1] The code for TVReg is available from `http://www2.compute.dtu.dk/~pcha/TVReg/`.

where $\mathrm{TV}(\boldsymbol{x})$ uses standard finite differences to compute the gradient magnitude. The segmentation in the classical approach is done by means of a Markov random field algorithm via graph cuts [9] using the `graph-tool` software [29], assuming that the number K of classes is known.

6.5.1 Test Using Artificial Data

The test problem used throughout these experiments is a standard 2D CT problem with 58 projections at angles $(i/58) \cdot 180°$, $i = 1, 2, \ldots, 58$, and with 181 parallel rays for each angle, and the image is 128×128 pixels. The corresponding sparse matrix A is thus of size $M \times N = 10,499 \times 16,384$ corresponding to an underdetermined system. Each nonzero element a_{ij} is the length of ray i through pixel j, and A was generated by means of the function `paralleltomo` from the MATLAB package[2] AIR Tools [17]. Given the exact test image $\boldsymbol{x}^{\text{exact}}$ we generate data with additive Gaussian noise as $\boldsymbol{b} = A\,\boldsymbol{x}^{\text{exact}} + \boldsymbol{e}$, where each element of the noise vector $\boldsymbol{e}$ is from the same normal distribution with zero mean and standard deviation chosen such that $\|\boldsymbol{e}\|_2 / \|A\,\boldsymbol{x}^{\text{exact}}\|_2 = 0.01$.

To study the performance of the algorithms, we use four different test images $\boldsymbol{x}^{\text{exact}}$ (they are shown in Figures 6.5–6.6):

1. The Shepp-Logan phantom from MATLAB.
2. A binary (2-class) phantom consisting of an image with random fine structures that are mostly horizontal.
3. A 4-class phantom with random regions separated by thin structures.
4. A gray-scale image with sharp contours and regions with smoothly varying intensities.

We include the Shepp-Logan phantom because it is so widely used (although it is an easy problem for TV regularization). The last other phantoms are generated by the function `phantomgallery` in AIR Tools.

In the FBP reconstructions we used the Hann filter and linear interpolation. In the TV reconstruction algorithm and in our method we manually chose the

[2] Version 1.2 of the AIR Tools package is available from `http://www2.compute.dtu.dk/~pcha/AIRtools/`.

Table 6.1: Regularization parameters used in the SRS algorithm for the different test problems in this section and the real-data example in the next section.

		Shepp-Logan	Binary	4-class	Gray-scale	Real data
SRS-Tik	λ_{noise}	$4.2 \cdot 10^{-3}$	$4.5 \cdot 10^{-4}$	$6.5 \cdot 10^{-4}$	$4.0 \cdot 10^{-4}$	0.2
	λ_{class}	1.0	0.5	0.5	0.4	0.3
SRS-TV	λ_{noise}	$2.0 \cdot 10^{-2}$	$9.5 \cdot 10^{-4}$	$3.0 \cdot 10^{-2}$	$8.5 \cdot 10^{-4}$	–
	λ_{class}	0.4	0.3	0.32	0.5	–

regularization parameters that minimize the image reconstruction error. In the SRS algorithm we choose the regularization parameters to give a good balance between reconstruction errors and segmentation errors; the parameters we use are listed in Table 6.1. The development of an automatic procedure for choosing these parameters is outside the scope of this paper. The number of iterations n_1 in Stage 1 is based on the stopping rule (6.16); in Stage 2 we always used $n_2 = 5$ iterations.

6.5.1.1 Study of Convergence

Given the reconstructed image $\boldsymbol{x}^*$, the *relative image reconstruction error* is defined as

$$\varepsilon_{\text{rec}} = \|\boldsymbol{x}^{\text{exact}} - \boldsymbol{x}^*\|_2 / \|\boldsymbol{x}^{\text{exact}}\|_2. \tag{6.19}$$

Given the vector $\boldsymbol{s}^*$ of labels computed by means of (6.4), the *relative segmentation error* is defined as the fraction of mislabeled pixels,

$$\varepsilon_{\text{seg}} = \frac{1}{N} \sum_{j=1}^{N} I(s_j^{\text{exact}} \neq s_j^*), \tag{6.20}$$

where $\boldsymbol{s}^{\text{exact}}$ is the vector of true labels, N is the number of pixels in the image, and I is a logical indicator function.

The error histories for algorithm SRS-Tik are shown in Fig. 6.3, and images and segmentations for selected iterations are shown in Fig. 6.4 (the error histories, reconstructions, and segmentations for SRS-TV are very similar and not shown here). The error decreases monotonically; the first few iterations give the biggest reduction and then the error reaches a plateau. During Stage 1 both the reconstruction and the segmentation improve – the image intensities improve, noise

Table 6.2: Summary of errors. In each entry, the top and bottom numbers are the relative image reconstruction error ε_{rec} and the relative segmentation error ε_{seg}, respectively.

Test problem	FBP→Seg	TV→Seg	SRS-Tik	SRS-TV
1: Shepp-Logan	0.34	0.038	0.021	0.023
	0.056	0.0038	0.0026	0.0031
2: Binary	0.46	0.33	0.18	0.26
	0.096	0.035	0.015	0.029
3: 4-class	0.39	0.16	0.047	0.055
	0.38	0.077	0.0057	0.0064
4: Gray-scale	0.24	0.082	0.060	0.087
	0.095	0.0040	0.0047	0.0051

is removed, and the edges become increasingly sharper. A few steps of Stage 2 improves the quality of the reconstruction, while the segmentation is hardly changed.

6.5.1.2 Study of Reconstruction and Segmentation Errors

The reconstruction and segmentation errors ε_{rec} and ε_{seg} for all algorithms and phantoms are summarized in Table 6.2, and the computed images and segmentations are shown in Figures 6.5 and 6.6. "FBP→Seg" and "TV→Seg" denote reconstruction by means of FBP and TV, respectively, followed by segmentation, while "SRS-Tik" and "SRS-TV" denote our algorithm with the regularization terms (6.9) and (6.8), respectively.

We see from both the table and the figures that the worst results are produced by FBP → segmentation; both the reconstructed image and the segmented image contain a lot of noise and artifacts.

The results for TV → segmentation are closer to the ground truth, but the reconstructions and the segmentations have the characteristic TV artifacts: the reconstructions capture small details but they also exhibit the standard TV staircasing effect, they include a number of small unwanted "clusters" of pixels, and the pixel intensities tend to be incorrect. Indeed, the images of the pixel-wise absolute errors $|\boldsymbol{x}^{\text{exact}} - \boldsymbol{x}^*|$ clearly show the incorrect image intensities with errors in the range 0.1–0.4. The underlying problem is that we underestimate jumps in the image intensities [34].

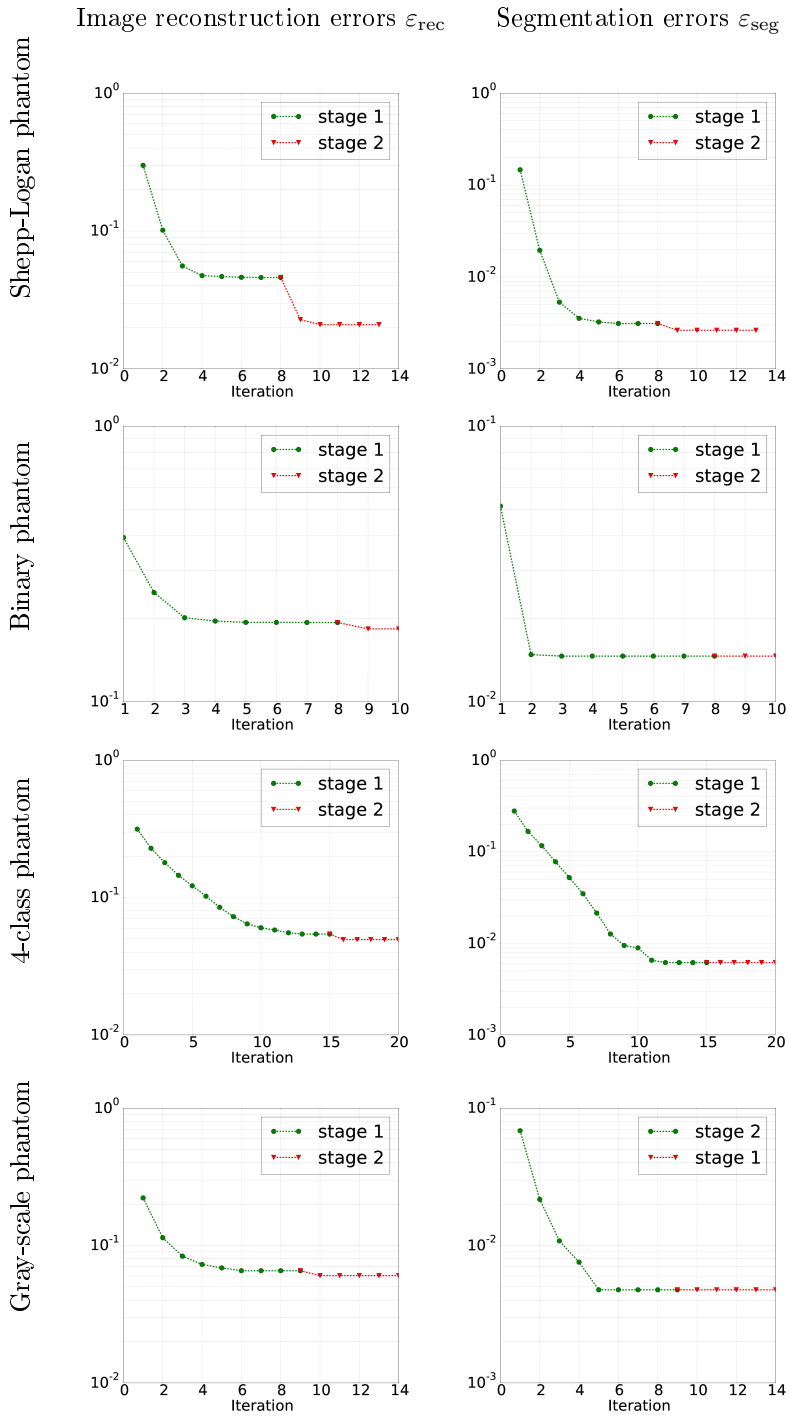

Figure 6.3: Error histories: the evolution of the image and segmentation errors ε_{rec} and ε_{seg} during the iterations of the SRS-Tik algorithm (the error histories for SRS-TV are very similar).

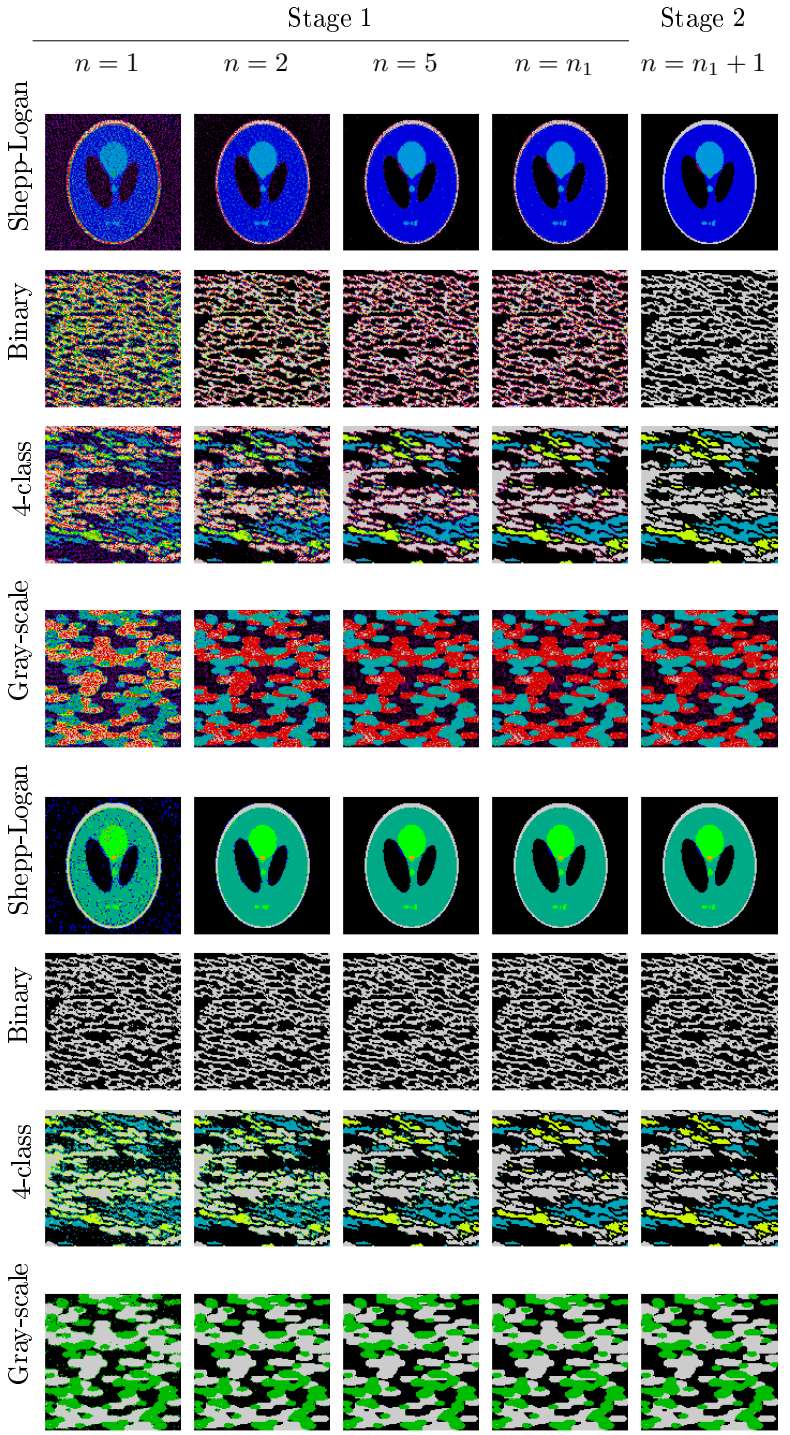

Figure 6.4: Images $\boldsymbol{x}^n$ (top) and segmentations $\boldsymbol{s}^n$ (bottom) for selected iterations during the SRS-Tik algorithm. For colorbars, see Figs. 6.5 and 6.6.

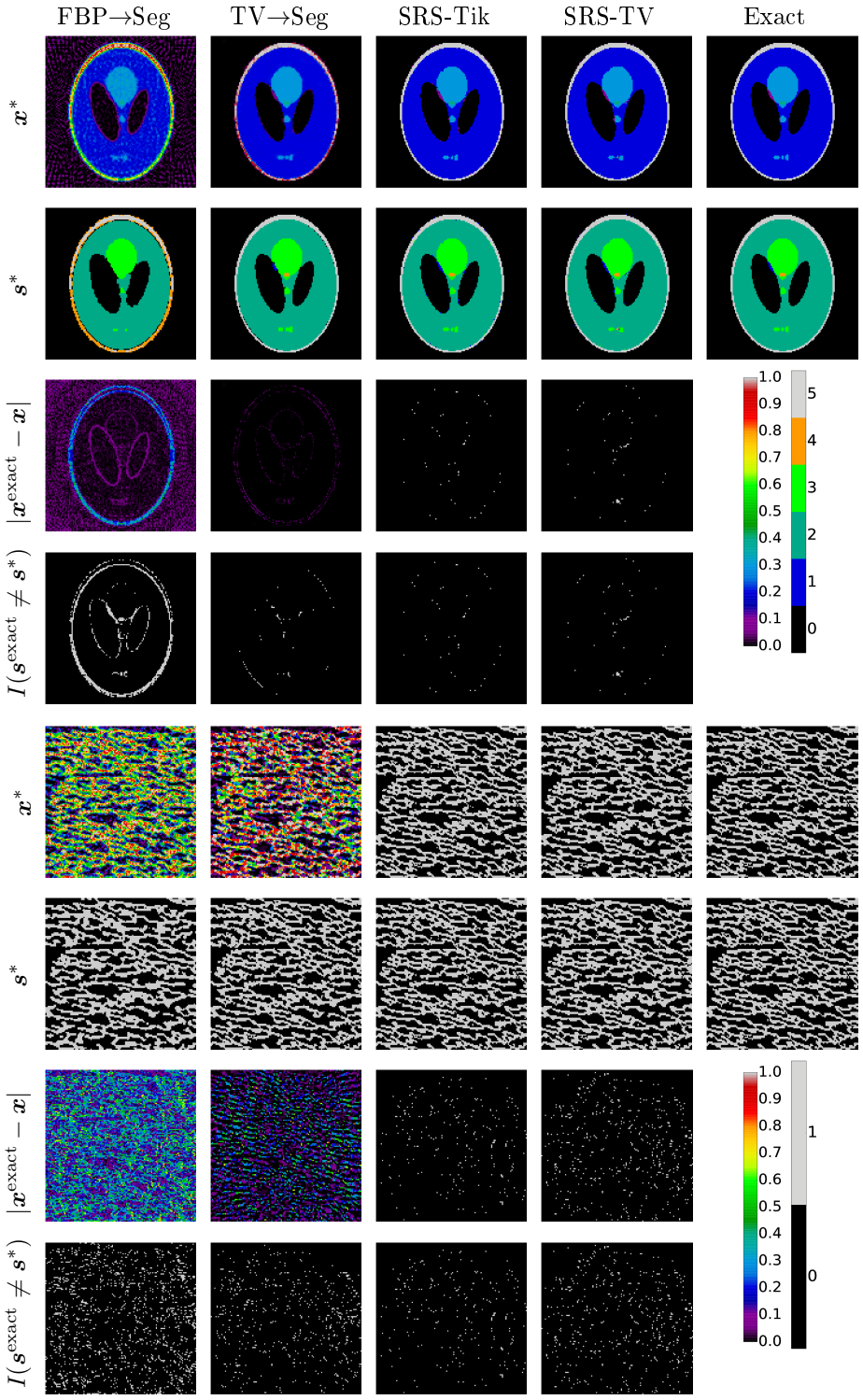

Figure 6.5: Results for the Shepp-Logan test problem (four top rows) and the binary test problem (bottom four rows). The left and right color-bars are for the reconstructions and segmentations, respectively.

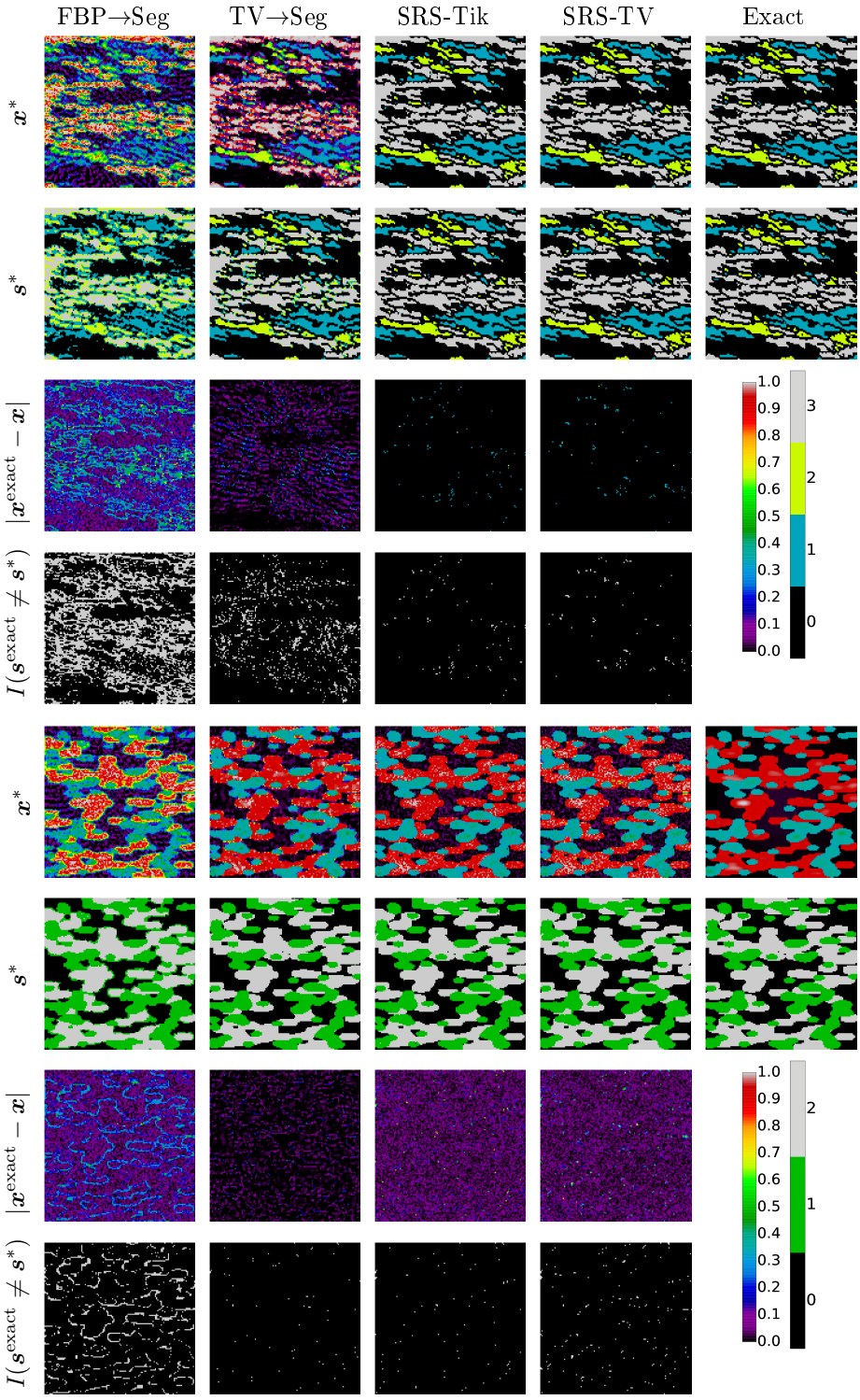

Figure 6.6: Results for 4-class test problem (top four rows) and the gray-scale test problem (bottom four rows). The left and right colorbars are for the reconstructions and segmentations, respectively.

We note that the TV reconstructions are slightly under-regularized, i.e., they include some amount of noise. Our empirical experience is that a bit of under-regularization in TV give less segmentation errors, because a slightly under-regularized TV reconstruction is more likely to include small features of the ground truth image.

For test problems 1, 2 and 3 the results from our SRS algorithm are more accurate than those from the classical approach, both for the reconstructions and the segmentations. This is especially true for test problems 2 and 3 which contain a lot of fine-structure, and for all test problems our reconstructions look significantly sharper than those from FBP and TV. Moreover we avoid the staircasing effect associated with TV reconstructions.

From a computational point of view, the TV regularization term R_{TV} (6.8) in algorithm SRS-TV is more difficult to handle than the Tikhonov term R_{Tik} (6.9) used in SRS-Tik. As a consequence, for SRS-TV the range of appropriate regularization parameters is quite small and the computing time is large, while for SRS-Tik the range for appropriate parameters is larger and the computing time is significantly smaller. From a practical point of view this makes the SRS-Tik algorithm more favorable.

Another advantage of SRS-Tik over SRS-TV is that during the iterations the former allows more flexibility in the location of sharp transitions in the probabilities in the HMMFM. This is because TV insists on putting discontinuities in these probabilities, while Tik allows more smooth transitions, and hence the location of the edges is done in a more flexible way. This is particularly evident from test problem 3, see the images during the iterations shown in Fig. 6.4. That means that in the pixels that are close to the detected edge the σ_j is rather big and that allows to tune the edge better during the iterations.

In conclusion we propose to use the Tikhonov regularization term and the corresponding SRS-Tik algorithm, because it requires less computing time, the choice of the regularization parameter is less critical, and its reconstructions and segmentations are competitive with those of algorithm SRS-TV.

6.5.2 Robustness Test

In our algorithm we assume that the user specifies the parameters μ_k and σ_k. It is interesting to see how sensitive the computed reconstruction and segmentation

Table 6.3: Sensitivity experiments with the 4-class phantom and the SRS-Tik algorithm. The true estimates are $\mu_1 = 0$, $\mu_2 = 0.33$, $\mu_3 = 0.66$, $\mu_4 = 1$ and the results in the previous section were obtained with $\sigma_k = 10^{-4}$, $k = 1, 2, 3, 4$. For these values the errors are $\varepsilon_{\text{rec}} = 0.047$ and $\varepsilon_{\text{seg}} = 0.0057$, cf. Table 6.2. See the text for an explanation of the experiments.

Experiment	μ	0.5	0.55	0.6	0.7	0.75	0.8
1	ε_{rec}	0.14	0.106	0.074	0.056	0.095	0.139
	ε_{seg}	0.05	0.027	0.013	0.007	0.023	0.06
2	ε_{rec}	0.27	0.247	0.165	0.082	0.122	0.151
	ε_{seg}	0.146	0.130	0.065	0.018	0.040	0.076
3	ε_{rec}	0.132	0.103	0.083	0.067	0.105	0.142
	ε_{seg}	0.042	0.025	0.016	0.011	0.033	0.065

are to the estimates μ_k of the attenuation coefficients and the associated σ_k. To study this we changed one of the estimates to a wrong value and analyzed how the resulting reconstruction and segmentation change, compared to results obtained with the correct estimate.

The results are shown in Table 6.3 for the SRS-Tik algorithm with the same choice of λ_{noise} and λ_{class} as before applied to the 4-class phantom. In all the experiments we kept μ_1, μ_2 and μ_4 at the correct values and varied μ_3 around its correct value 0.66. We show the reconstruction and segmentation errors (6.19) and (6.20) for the following three experiments:

1. Here we kept $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 10^{-4}$.
2. Here $\sigma_1 = \sigma_2 = \sigma_4 = 10^{-4}$ while $\sigma_3 = 2 \cdot 10^{-4}$.
3. Here $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = 2 \cdot 10^{-4}$.

When μ_3 assumes an incorrect value then clearly the errors increase, but we note that the results are still better than the results obtained with the TV→Seg algorithm for which the errors are $\varepsilon_{\text{rec}} = 0.16$ and $\varepsilon_{\text{seg}} = 0.077$ (from Table 6.2). Hence, our algorithm can handle an incorrect estimate as long as it does not take a value close to one of the other estimates (in which case the segmentation becomes wrong).

One might expect that we can compensate for a wrong estimate μ_3 with a bigger σ_3. Unfortunately, this is not true for our algorithm as documented in the table.

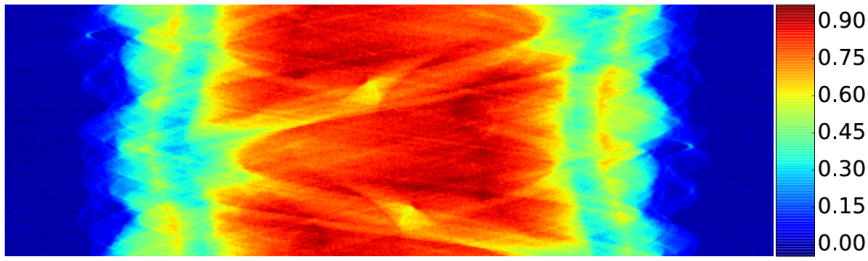

Figure 6.7: The measured data (the sinogram) for the real-data problem.

In fact, changing only one σ_k tends to increase the errors, and in particular the segmentation error, because many pixels are mislabel when the parameters σ_k are unbalanced. As also shown in the table, increasing all σ_k to the same value is preferable, and we note that we obtain almost the same errors when all σ_2 are twice as large.

The conclusion of this experiment is that we should use balanced values for σ_k and that we cannot use σ_k to compensate for an incorrect μ_k.

6.5.3 Test Using Real Data

We complete our performance studies with some experiments using real data. The data was collected using a micro-CT scanner with fan-beam geometry that scans the object slice-by-slice. For our reconstruction we use 167 projections, with 512 rays in each projection. The reconstructed image contains 362×362 pixels, and hence the amount of pixels is approximately 1.5 times larger than the amount of data. The data, in the form of the sinogram, is shown in Fig. 7.3.

The scanned object is a candy that contains air, a nut at the center, pieces of nuts, chocolate, and waffle, enclosed in a thin aluminum foil. Nuts and waffle have almost the same attenuation coefficients, so we treat them as the same class. The foil, which has a high attenuation coefficient, is very thin and since the resolution is too low to accurately represent the foil, the reconstructed coefficients are inaccurate; for this reason we do not account for the foil in our classes. As priors for the classes we used the following mean values and standard deviations.

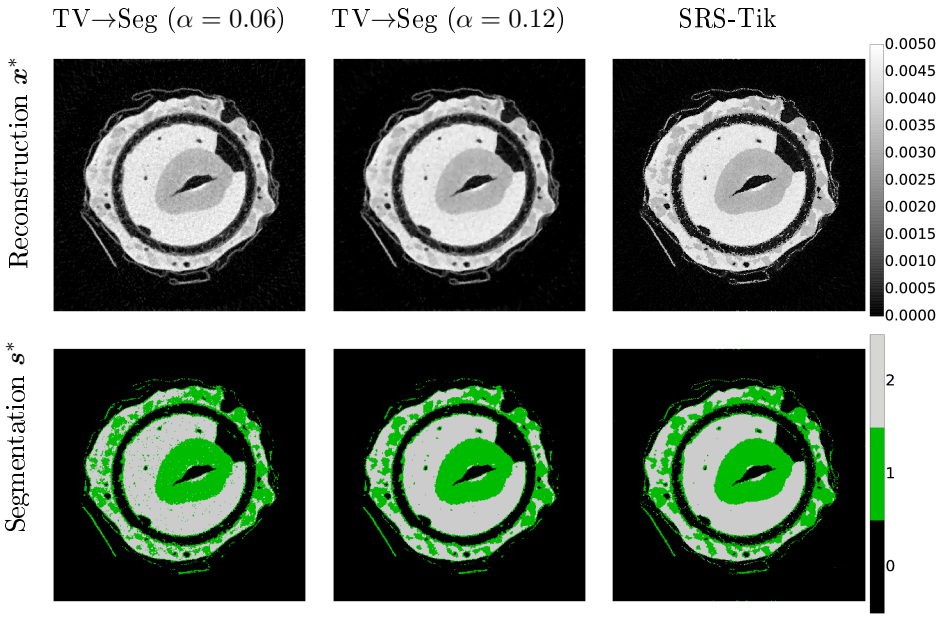

Figure 6.8: Results for the real-data problem; all reconstructs/segmentations have the same color scale.

- Air: $\mu_1 = 0$, $\sigma_1 = 0.001$.
- Nuts and waffle: $\mu_2 = 0.0033$, $\sigma_2 = 0.001$.
- Chocolate: $\mu_3 = 0.0044$, $\sigma_3 = 0.001$.

Knowing that FBP is not well suited for underdetermined problems, we compare two algorithms: TV $\rightarrow$ segmentation (using again the alpha-expansion segmentation algorithm) and our SRS-Tik algorithm. The matrix A for this problem was generated with the function `fanbeamtomo` from AIR Tools [17].

The reconstructions are shown in Fig. 6.8. We computed TV reconstructions for two different values of the regularization parameter α in (6.18); for the small value we are able to recover small details in the segmentation, but there are many misclassified pixels; for the larger parameter we have more homogeneous areas in the segmentation but we miss many small details (for example, it is difficult to see the edges between different phases). In conclusion, the TV $\rightarrow$ segmentation algorithm does not perform so well.

In the SRS-Tik reconstruction the edges are always clearly visible and overall image looks significantly sharper than TV reconstruction. Although our reconstruction looks slightly more noisy (because we do not use a smoothing prior in the reconstruction), we emphasize that the majority of the noise-looking pixels are actually correlated with the texture of the reconstructed object. Our segmentation of the object identifies more correct pixels than the TV $\rightarrow$ segmentation algorithm.

Although we do not know the foil's precise attenuation coefficient, in the SRS-Tik reconstruction the position of the foil is reconstructed correctly and better than in case TV$\rightarrow$Seg algorithm – in both algorithms it is assigned to the second class.

Bibliography

[1] Kyongtae T Bae et al. "Automatic Segmentation of Liver Structure in CT Images". In: *Medical Physics* 20.1 (1993), pp. 71–78.

[2] Johnathan M Bardsley and James G Nagy. "Covariance-Preconditioned Iterative Methods for Nonnegatively Constrained Astronomical Imaging". In: *SIAM Journal on Matrix nalysis and Applications* 27.4 (2006), pp. 1184–1197.

[3] K. J. Batenburg and J. Sijbers. "DART: a Practical Reconstruction Algorithm for Discrete Tomography". In: *IEEE Trans. Image Process.* 20.9 (2011), pp. 2542–2553.

[4] K. J. Batenburg et al. "DART: a Robust Algorithm for Fast Reconstruction of Three-Dimensional Grain Maps". In: *J. Appl. Cryst.* 40 (2010), pp. 1464–1473.

[5] Dimitri P. Bertsekas. *Nonlinear Programming, 2. Ed.* Athena Scientific, Belmont, MA, 1999.

[6] Junguo Bian et al. "Evaluation of Sparse-View Reconstruction from Flat-Panel-Detector Cone-Beam CT". In: *Physics in Medicine and Biology* 55.22 (2010), p. 6575.

[7] Åke Björck. *Numerical Methods for Least Squares Problems.* SIAM, Philadelphia, 1996.

[8] Yuri Boykov and Gareth Funka-Lea. "Graph Cuts and Efficient ND Image Segmentation". In: *Int. J. Comp. Vis.* 70.2 (2006), pp. 109–131.

[9] Yuri Boykov, Olga Veksler, and Ramin Zabih. "Fast Approximate Energy Minimization via Graph Cuts". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 23.11 (2001), pp. 1222–1239.

[10] Yuri Y Boykov and M-P Jolly. "Interactive Graph Cuts for Optimal Boundary and Region Segmentation of Objects in ND Images". In: *Proceedings of the Eighth IEEE International Conference on Computer Vision*. Vol. 1. IEEE. 2001, pp. 105–112.

[11] Ronald Newbold Bracewell and ACf Riddle. "Inversion of Fan-Beam Scans in Radio Astronomy". In: *The Astrophysical Journal* 150 (1967), p. 427.

[12] Thomas Brox et al. "Colour, Texture, and Motion in Level Set Based Segmentation and Tracking". In: *Image and Vision Computing* 28.3 (2010), pp. 376–390.

[13] TM Buzug. *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*. Springer, NY, 2010.

[14] Maarten Depypere et al. "The minimal entropy prior for simultaneous reconstruction and segmentation of in vivo microCT trabecular bone images". In: *Biomedical Imaging: From Nano to Macro, 2009. ISBI'09. IEEE International Symposium on*. IEEE. 2009, pp. 586–589.

[15] Yi Gao et al. "Sparse Texture Active Contour". English. In: *IEEE Transactions on Image Processing* 22.10 (2013), pp. 3866–3878. ISSN: 10577149, 19410042. DOI: 10.1109/TIP.2013.2263147.

[16] Per Christian Hansen. *Discrete inverse problems: insight and algorithms*. Vol. 7. Siam, 2010.

[17] Per Christian Hansen and Maria Saxild-Hansen. "AIR tools—a MATLAB package of algebraic iterative reconstruction methods". In: *Journal of Computational and Applied Mathematics* 236.8 (2012), pp. 2167–2178.

[18] Tobias Lindstrøm Jensen et al. "Implementation of an Optimal First-Order Method for Strongly Convex Total Variation Regularization". In: *BIT Numer. Math.* 52.2 (2012), pp. 329–356.

[19] Avinash C.. Kak and Malcolm Slaney. *Principles of computerized tomographic imaging*. Society for Industrial and Applied Mathematics, 2001.

[20] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. "Snakes: Active contour models". In: *International journal of computer vision* 1.4 (1988), pp. 321–331.

[21] E. Klann. "A Mumford-Shah-like Method for Limited Data Tomography with an Application to Electron Tomography". In: *SIAM J. Imaging Sciences* 4 (2011), pp. 1029–1048. DOI: 10.1137/100817371.

[22] E. Klann, R. Ramlau, and W. Ring. "A Mumford-Shah Level-Set Approach for the Inversion and Segmentation of SPECT/CT Data". In: *Inverse Problems and Imaging* 5 (2011), pp. 137–166. DOI: 10.3934/ipi.2011.5.137.

[23] Julien Mairal et al. "Discriminative Learned Dictionaries for Local Image Analysis". In: *IEEE Conference on Computer Vision and Pattern Recognition, 2008.* 2008, pp. 1–8.

[24] Jose L Marroquin, Edgar Arce Santana, and Salvador Botello. "Hidden Markov measure field models for image segmentation". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25.11 (2003), pp. 1380–1387.

[25] A Mohammad-Djafari. "Gauss-Markov-Potts Priors for Images in Computer Tomography Resulting to Joint Optimal Reconstruction and Segmentation". In: *International Journal of Tomography and Statistics* 11 (2009), pp. 76–92.

[26] Eric N Mortensen and William A Barrett. "Interactive Segmentation with Intelligent Scissors". In: *Graphical Models and Image Processing* 60.5 (1998), pp. 349–384.

[27] JL Mueller and S Siltanen. *Linear and Nonlinear Inverse Problems with Practical Applications.* SIAM, Philadephia, 2012.

[28] Stanley Osher and Nikos Paragios. *Geometric level set methods in imaging, vision, and graphics.* Springer Science & Business Media, 2003.

[29] Tiago P. Peixoto. "The Graph-tool Python Library". In: (2014). DOI: 10.6084/m9.figshare.1164194. URL: http://figshare.com/articles/graph_tool/1164194 (visited on 09/10/2014).

[30] R. Ramlau and W. Ring. "A Mumford-Shah Level-Set Approach for the Inversion and Segmentation of X-ray Tomography Data". In: *J. Comput. Phys.* 221 (2007), pp. 539–557. DOI: 10.1016/j.jcp.2006.06.041.

[31] James Albert Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science.* Cambridge University Press, 1999.

[32] Emil Y Sidky, Chien-Min Kao, and Xiaochuan Pan. "Accurate Image Reconstruction from Few-Views and Limited-Angle Data in Divergent-Beam CT". In: *Journal of X-ray Science and Technology* 14.2 (2006), pp. 119–139.

[33] Dominique Van de Sompel and Michael Brady. "Simultaneous reconstruction and segmentation algorithm for positron emission tomography and transmission tomography." In: *ISBI.* 2008, pp. 1035–1038.

[34] David Strong and Tony Chan. "Edge-Preserving and Scale-Dependent Properties of Total Variation Regularization". In: *Inverse Problems* 19.6 (2003), S165.

[35] Layer T et al. "PET Image Segmentation Using a Gaussian Mixture Model and Markov Random Fields". In: *EJNMMI Physics* 2:9 (2015). DOI: 10.1186/s40658-015-0110-7.

[36] Sungwon Yoon, Angel R. Pineda, and Rebecca Fahrig. "Simultaneous Segmentation and Reconstruction: A Level Set Method Approach for Limited View Computed Tomography". In: *Med. Phys.* 37.5 (2010), pp. 2329–2340.

Chapter 7

Relaxed Simultaneous Tomographic Reconstruction and Segmentation with Class Priors for Poisson Noise

This work is a continuation of work on algorithms for simultaneous reconstruction and segmentation. In our previous work we developed an algorithm for data with Gaussian noise, and in that algorithm the coefficient matrix for the system is explicitly store. We improve this algorithm in two ways: our new algorithm can handle Poisson noise in the data, and it can solve much larger problems since it does not store the matrix. We formulate this algorithm and test it on artificial test problems. Our results show that the algorithm performs well, and that we are able to produce reconstructions and segmentations with small errors.

7.1 Introduction

In this work we continue developing the method of Simultaneous Reconstruction and Segmentation (SRS) that makes reconstruction and segmentation in joint fashion that first was presented in the work [22] and was further developed in [19]. The main goal of this work is to solve larger problems and to apply the algorithm to problems with Poisson noise. We will call our method in this work SRS-II as it is different from the one we presented in the first work. To make the search for a minimum of a non-convex function more robust, we use approach similar to the Simulated Annealing approach. We test our approach on an artificial problem.

There is a variety of methods that allow to make reconstruction. The main analytical method is Filtered Back Projection (FBP), also known as inverse Radon Transform [7], [13]. The strong sides of this algorithm are simplicity and low computational time, but many projections are usually needed for a good reconstruction. In case the amount of data that is provided for the reconstruction is not enough, the results will be poor. Another drawback of this algorithm is that it is significantly affected by noise. In case the data that was collected is not very precise - the reconstruction may be noisy. This drawback in many cases may be compensated by amount of data, but collecting big amounts of data in many cases is unwanted or expensive. One more drawback of this approach is that the supported geometries are very limited.

Another class of important techniques are the algebraic reconstruction techniques [11]. This includes Kaczmarz method [12], Cimmino method [16] and many others. These methods rely on the phenomenon of semiconvergence [10]. The main advantage of these techniques is that they support any reconstruction geometry as long as it may be formulated in the form of a set of linear system of equations $\mathbf{b} = A\mathbf{x}$, where A is a matrix, $\mathbf{x}$ is a vector that represents object, $\mathbf{b}$ is the vector that represents the measurements. These methods are easy to use and well researched. There are several drawbacks of these algorithms. One of them is that these methods usually do not take into account the specific types of the noise that may appear during the measurement process. Another drawback is that these methods usually are not noise robust: small deviations in the data may lead to big deviations in solutions in case the problem is underdetermined.

Next, quite important set of methods are variational techniques [3], [21]. These techniques are based on well-known and widely used optimization algorithms. The main idea of this approach is to formulate the reconstruction problem as an optimization problem and after that to solve this problem with any of available

optimization technique. To solve the optimization problem fast and precise, usually, the problem should be convex and the function under optimization should have an analytical gradient. This approach is significantly more flexible than algebraic reconstruction techniques. As the problem may be solved as soon as it is convex, the formulated problem may be quite complex. The method that is described below belongs to this class of the approach.

One of the most interesting variation of this approach for us is the Total Variation (TV) reconstruction method [9]. It's main idea is to maximize the likelihood of observing the data that was received during the measurements (that in the future we will call data fitting term) and, in the same time, to minimize the integral of the norm of the gradient in the image (this is usually referred to as a regularization term). The advantages of this approach are: simplicity, usage of well researched optimization techniques, predictable behaviour, noise robustness. Also, to make a good reconstruction with this method less data is needed than with all the methods that were described above. One of the main aims of this method is to prevent smoothening of the edges. As for the disadvantages, this methods in many cases smears out some important small details. Also, the staircasing is a well-known artifact of this method (usually happens on the edges of the object or on the regions with a gradient, substitutes the gradient or the edge with a set of steps with different intensity that is the same inside of the step) [23].

The Monte-Carlo approaches to these problems are able to find the global minimum of the objective function. It solves non-convex problems, but the long time of computing the solution is a significant drawback of this approach. Because of that this is very rarely used to solve real problems. As an example of a good problem for this approach is a reconstruction with l_0 norm of the gradient of the image that corresponds to penalty of the length of the edges in the image. We do not consider this approach as a good candidate to solve our problem.

As well as for the reconstruction, for segmentation there exist many approaches to do a segmentation. One of the main direction here is called snakes [14]. This approach needs manual input and thus is not good for our application as we want to register the regions automatically without any interaction with human.

Another approach in this area is based on Markov Random Fields (MRF) such as Potts model, where for each of the pixels (or in some applications, regions) one label is assigned as a class to which the pixel belongs. Usually in this approach the algorithm that is used to get the results is Graph-Cut method [5], [4], [6]. In case of more than two labels classification, Alpha-Expansion algorithm (modification of Graph-Cut method) is used, although this approach gives

only the approximation to the solution as overall problem in case of non-binary segmentation is NP-hard as the problem is discrete.

Finally, another popular set of segmentation methods is called Level-Set methods[20], [17], [1]. These methods are real-valued methods that use convex optimization techniques to find the segmentation. The advantage of these methods is that in case of well-formulated (i.e., convex and has an analytical gradient) problem, it will output the only minimum with given precision. Also, for the majority of the optimization techniques, the dependency of computational time on precision is well-known.

In our work as a model for a segmentation we use a Hidden Markov Measure Field Models (HMMFM) [15]. The main idea of the HMMFM is to assign to each of the pixels the probability with which it belongs to each of the classes. Thus, the HMMFM may be considered as a relaxation of the MRF. The problem of computing the optimal HMMFM may be easily formulated as an optimization problem with a regularization term and may be easily solved as soon as the problem is convex.

Usually in the applications one does reconstruction first and after that makes a segmentation of the received segmentation. This approach itself usually gives good results, but in case the expected classes are known, does not utilize all the available information to generate a good reconstruction output. More than that, in case some errors occur on the reconstruction step - they are likely to propagate to the segmentation of the reconstruction.

We would like to improve the results of both reconstruction and segmentation by utilizing information about the classes of materials that are likely to exist in the object and by incorporating the segmentation and the reconstruction into one joint procedure.

To accomplish this we use variational approach for both reconstruction and segmentation problems. As a model for segmentation we use Hidden Markov Measure Field Models (HMMFM). This makes our problem a constrained optimization problem. We use a statistical approximation to make this problem convex. After that we are able to apply standard optimization techniques to compute reconstruction and segmentation. For the image optimization we use well-known L-BFGS algorithm [8] and for HMMFM optimization we use Frank-Wolfe algorithm [2].

Table 7.1: A simple example of the HMMFM; the numbers in the table are the probabilities in the HMMFM for each class and each pixel.

class k	pixel 1	pixel 2	pixel 3	pixel 4	$\cdots$	pixel j	$\cdots$	pixel N
1	0.1	0.2	0.2	0.3	$\cdots$	δ_{j1}	$\cdots$	0.9
2	0.0	0.1	0.2	0.0	$\cdots$	δ_{j2}	$\cdots$	0.0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$		$\vdots$
K	0.8	0.6	0.3	0.0	$\cdots$	δ_{jk}	$\cdots$	0.1
$\sum_{k=1}^{K} \delta_{jk}$	1.0	1.0	1.0	1.0	$\cdots$	1.0	$\cdots$	1.0

7.2 Problem formulation

We solve the problem of reconstruction of an image $\mathbf{x}$ given the set of projections $\mathbf{b}$. We denote the values of the pixel j of the image as x_j and the recorded projection i value as b_i. The projection matrix is denoted as A in the matrix form and the element that encodes the length of the ray i inside of the pixel j is denoted as a_{ij}. The projections $\mathbf{b}$ and the projection matrix A are known. Thus, the forward model can be formulated as

$$A\mathbf{x} = \mathbf{b}. \tag{7.1}$$

We consider the noise to be Poisson noise.

We start with the summary of the theory from our previous work and then develop it for our new problem.

We assume the classes of materials that could be found in the reconstructed image known. We consider that the classes have Gaussian distribution of the attenuation coefficients with the mean values μ_k and standard deviations σ_k, where k is the index of the class.

To do the segmentation of the image we use the concept of the Hidden Markov Measure Field Model (HMMFM). We denote the HMMFM as matrix δ. The HMMFM can be represented as a table of probabilities of the pixel to belong to the specific class. The HMMFM illustration can be found on the Table 7.1. The value of the HMMFM for the pixel j and class k we denote as δ_{jk}. Thus, the

HMMFM over the classes should sum up to one in each pixel:

$$\begin{aligned} \forall\ j\ \sum_k \delta_{jk} = 1; \\ \forall\ j,k\ \delta_{j,k} \geq 0. \end{aligned} \tag{7.2}$$

It is easy to formulate the problem of reconstructing $\mathbf{x}$ and δ from $\mathbf{b}$ in terms of probability maximization:

$$\begin{aligned} \mathbf{x}^*, \delta^* = &\arg\max_{\mathbf{x},\delta} p(\mathbf{x}, \delta|\mathbf{b}) \\ \text{s.t.} \quad &\forall\ j\ \sum_k \delta_{jk} = 1; \\ &\forall\ j,k\ \delta_{j,k} \geq 0 \\ &\forall j \quad x_j > 0 \end{aligned} \tag{7.3}$$

This problem can be reformulated using Bayes rule:

$$p(\mathbf{x}, \delta|\mathbf{b}) = \frac{p(\mathbf{b}|\mathbf{x}, \delta)p(\mathbf{x}|\delta)p(\delta)}{p(\mathbf{b})}. \tag{7.4}$$

Here the probability of the measured data given the HMMFM δ and the image $\mathbf{x}$ can be written as $p(\mathbf{b}|\mathbf{x})$ because the data $\mathbf{b}$ depends only on the image $\mathbf{x}$. Due to the positivity of the probability and the monotonicity of the logarithm, the problem (7.3) can be represented as a maximization problem of the logarithm of the probability $p(\mathbf{x}, \delta|\mathbf{b})$:

$$\log p(\mathbf{x}, \delta|\mathbf{b}) = \log p(\mathbf{b}|\mathbf{x}) + \log p(\mathbf{x}|\delta) + \log p(\delta) - \log p(\mathbf{b}). \tag{7.5}$$

The probability $p(\mathbf{b})$ does not depend on variables under optimization ($\mathbf{x}$ and δ). Therefore, the optimization problem without the last term and with it have the same optimal point meaning that we can consider the problem without it:

$$\log p(\mathbf{x}, \delta|\mathbf{b}) = \log p(\mathbf{b}|\mathbf{x}) + \log p(\mathbf{x}|\delta) + \log p(\delta) + \text{const}. \tag{7.6}$$

In this formula the first term is a data fidelity term, the goal of this term is to fit the data the best possible way. In this work we consider only the problems, where the projections are obtained via measurements of the photons. In this problem the measured values are distributed according to the Poisson distribution:

$$p(\mathbf{b}|\mathbf{x}) = \prod_i \frac{\lambda_i^{b_i} \exp(-\lambda_i)}{b_i!}, \tag{7.7}$$

where the λ_i are the expected value of the measured values on the detector i. In case of the Emission Tomography the Poisson distribution has the following expected values for the measurements:

$$\lambda_i = \sum_j a_{ij} x_j. \tag{7.8}$$

In our approach we maximize the logarithm of the probability instead of the probability itself. Taking logarithm of the equation (7.7), we get the following expression taking into account that $\log(b_i!)$ does not depend on the values λ_i and, thus, does not depend on $\mathbf{x}$ due to (7.8):

$$\log p(\mathbf{b}|\mathbf{x}) = \sum_j (b_i \log \lambda_i - \lambda_i) + \text{const}. \tag{7.9}$$

The second term $\log p(\mathbf{x}|\delta)$ in (7.6) is a class fidelity term. The probability under the logarithm is the product of the weighted mixture of Gaussians for each pixel, where weights are defined by the HMMFM, while the parameters σ_k, μ_k of the Gaussians are assumed to be known:

$$p(\mathbf{x}|\delta) = \prod_j \sum_k \delta_{jk} \frac{1}{\sqrt{2\pi\sigma_k}} \exp\left(-\frac{(x_j - \mu_k)}{2\sigma_k^2}\right). \tag{7.10}$$

The third term $\log p(\delta)$ of the sum in (7.6) is the statistical prior for the HMMFM. In this work we use the neighbourhood prior: in case a pixel is more likely to belong to a specific class, then the neighbours are encouraged to belong to the same class:

$$p(\delta) = \exp\left(-\Phi(\delta)\right). \tag{7.11}$$

Here the function $\Phi(\delta)$ is an l_2 norm of discrete approximation of the gradient for each of classes of the HMMFM:

$$\Phi(\delta) = \sum_{j,k} \sum_{j' \in N(j)} (\delta_{j,k} - \delta_{j',k})^2, \tag{7.12}$$

where $N(j)$ is a set of adjacent pixels of pixel j. In case of a 2D problem the $N(j)$ consists of the pixel that is 1 pixel above the pixel j and of the pixel that is 1 pixel to the left from the pixel j. From our experience the performance of different norms do not vary much, and l_2 norm has some advantages in terms of flexibility of segmentation and it is significantly easier to optimize.

Taking into account (7.4), (7.9), (7.10), (7.11), the overall problem (7.3) has the

following form:

$$\begin{aligned}\mathbf{x}^*, \delta = \arg\max_{\mathbf{x},\delta} \; & \lambda_1 \left[\sum_i b_i \log \sum_j a_{ij} x_j - \sum_i \sum_j a_{ij} x_j \right] + \\ & + \sum_j \log \sum_k \delta_{jk} \frac{1}{\sqrt{2\pi\sigma_k}} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right) - \\ & - \lambda_2 \Phi(\delta) \\ \text{s.t. } \forall j \quad & \sum_k \delta_{jk} = 1; \; \forall j, k \quad \delta_{jk} \geq 0 \\ & \forall j \quad x_j > 0 \end{aligned} \tag{7.13}$$

Here we introduce two regularization parameters λ_1, λ_2 that regulate how much the result is affected by the class fidelity and by neighborhood prior.

7.3 Simplification

The problem (7.13) is convex in terms of HMMFM δ, but is non-convex in terms of image $\mathbf{x}$. More than that, this problem for the majority of HMMFM realisations has many local minima. The main source of this non-convexity is the second term (7.10) of the optimization problem. To deal with this difficulty we make the substitution of the second term:

$$\tilde{p}(\mathbf{x}|\delta) = \prod_j \frac{1}{\sqrt{2\pi\tilde{\sigma}_j}} \exp\left(-\frac{(x_j - \tilde{\mu}_j)^2}{2\tilde{\sigma}_j} \right), \tag{7.14}$$

where the mean values and standard deviations are computed for each pixel:

$$\tilde{\mu}_j = \sum_k \delta_{j,k} \mu_k, \tag{7.15}$$

$$\tilde{\sigma}_j^2 = \sum_k \delta_{jk} (\sigma_k^2 + \mu_k^2) - \mu_j^2. \tag{7.16}$$

This simplification is convex in terms of $\mathbf{x}$, but non-convex in terms of δ. It is an approximation with a normal distribution of the distribution $p(\mathbf{x}|\delta)$ with the mean estimated with an expected value of this distribution and square of standard deviation estimated with variance of this distribution.

7.4 Algorithm

We minimize the problem iteratively and in the l-th iteration we would like to compute an approximation to the solution of the following problem:

$$\begin{aligned}\mathbf{x}^{l+1}, \delta^{l+1} = \arg\max_{\mathbf{x},\delta} \; & \lambda_1 \left[\sum_i b_i \log \sum_j a_{ij} x_j - \sum_{i,j} x_j a_{ij} \right] + \\ & + \sum_j \log \sum_k \delta_{jk} \frac{1}{\sqrt{2\pi\sigma_k}} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right) - \\ & - \lambda_2 \Phi(\delta) \\ \text{s.t. } \forall j \quad & \sum_k \delta_{jk} = 1; \; \forall j, k \quad \delta_{jk} \geq 0 \\ \forall j \quad & x_j > 0.\end{aligned} \tag{7.17}$$

Since to the problem is non-convex, we minimize the problem above using a two-stage algorithm. In the first stage we compute the approximation to the solution, while in the second stage we compute the final solution. In both stages we use the approximation to the second term of the problem to make the problem convex for the HMMFM.

7.4.1 First Stage

Since we have two variables to optimize, for one of which the problem is non-convex, we minimize this problem iteratively in two steps. In the first step we compute the approximation to the image $\mathbf{x}$ given the fixed HMMFM δ.

Thus, in the first step we would like to compute the solution to the part of the problem above that includes all the terms with the variable $\mathbf{x}$:

$$\begin{aligned}\mathbf{x}^{l+1} = \arg\max_{\mathbf{x}} \; & \lambda_1 \left[\sum_i b_i \log \sum_j a_{ij} x_j - \sum_{i,j} x_j a_{ij} \right] + \\ & + \sum_j \log \sum_k \delta_{jk}^l \frac{1}{\sqrt{2\pi\sigma_k}} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2} \right) \\ \text{s.t. } \forall j \quad & x_j > 0,\end{aligned} \tag{7.18}$$

but, as it was already mentioned, the second term here is non-convex. To deal with this non-convexity we use the approximation (7.14):

$$\begin{aligned}\mathbf{x}^{l+1} = \arg\max_{\mathbf{x}} \lambda_1 \left[\sum_i b_i \log \sum_j a_{ij}x_j - \sum_{i,j} x_j a_{ij}\right] + \\ + \sum_j \frac{(x_j - \tilde{\mu}_j^l)^2}{2(\tilde{\sigma}_j^l)^2} \\ \text{s.t. } \forall j \quad x_j > 0.\end{aligned} \tag{7.19}$$

With this approximation the problem becomes convex and we can easily minimize it using any optimization technique. We use a Limited Memory BFGS (L-BFGS) optimization algorithm [8] to minimize it. As we do many iterations of this two-steps algorithm - it is enough to do just few iterations of the L-BFGS - and we will get the next iteration of the image.

Although the problem (7.19) is convex, the overall problem is not. To deal with it we tried two approaches. One approach is to start with big data regularization λ_1 and gradually reduce it to a small value that was specified before the algorithm starts. Another approach is instead of λ_1 to gradually modify σ_k along the iterations.

We propose in the first approach to use the following expression for the regularization parameter:

$$\lambda_1^l = \lambda_1(1 + C\beta^l), \tag{7.20}$$

and for the second approach we propose to use the similar expression:

$$\sigma_k^l = \sigma_k(1 + C\beta^l), \tag{7.21}$$

where C and β is a constant. We require that $\lambda_1^l \to \lambda_1$ for $l \to \infty$ and $\sigma_k^l \to \sigma_k$ for $l \to \infty$. In order to achieve that, we should assign constant β to belong to the interval $(0, 1)$. In our experiments we use $C = 1000$, $\beta = 0.9$.

For the first approach the image reconstruction step will turn into the following problem:

$$\begin{aligned}\mathbf{x}^{l+1} = \arg\max_{\mathbf{x}} \lambda_1^l \left[\sum_i b_i \log \sum_j a_{ij}x_j - \sum_{i,j} x_j a_{ij}\right] + \\ + \sum_j \frac{(x_j - \tilde{\mu}_j^l)^2}{2(\tilde{\sigma}_j^l)^2} \\ \text{s.t. } \forall j \quad x_j > 0,\end{aligned} \tag{7.22}$$

for the second approach we need to recompute the values for $\tilde{\sigma}_j^l$ according to the rule 7.16, where σ_k should be substituted by σ_k^l. Then the reconstruction step turns into the following problem:

$$\begin{aligned}\mathbf{x}^{l+1} = \arg\max_{\mathbf{x}} \lambda_1 \left[\sum_i b_i \log \sum_j a_{ij} x_j - \sum_{i,j} x_j a_{ij} \right] + \\ + \sum_j \frac{(x_j - \tilde{\mu}_j^l)^2}{2(\tilde{\sigma}_j^l)^2} \\ \text{s.t. } \forall j \quad x_j > 0.\end{aligned} \tag{7.23}$$

In principle, we could iterate L-BFGS to convergence to the solution of (7.23), but since this computation is only one step of our iterations, it is enough to improve the solution by few iterations of L-BFGS algorithm.

Once the image is updated we should find the next iteration of the segmentation. In the full problem (7.17) we keep the $\mathbf{x}$ fixed and optimize δ:

$$\begin{aligned}\delta^{l+1} = \arg\max_{\delta} \sum_j \log \sum_k \delta_{jk} \frac{1}{\sqrt{2\pi\sigma_k}} \exp\left(-\frac{(x_j^{l+1} - \mu_k)^2}{2\sigma_k^2} \right) - \\ - \lambda_2 \Phi(\delta) \\ \text{s.t. } \forall j \quad \sum_k \delta_{jk} = 1; \ \forall j,k \quad \delta_{jk} \geq 0.\end{aligned} \tag{7.24}$$

This is a convex problem with constraints. The constraints of this problem are simplices and we minimize it with a modified Frank-Wolfe algorithm that is specially designed to minimize the problems with these constraints.

Again, it is possible to do these iterations until convergence, but it is sufficient to make small improvement of the result.

7.4.2 Second Stage

In our original algorithm we also included a second stage. In this work we do not implement this approach, but we still want to formulate it. The purpose of the second stage is to force the solution to be closer to the class prior.

The second stage of the algorithm is very similar to the first stage: it consists of iterations that include the image optimization and HMMFM optimization using

the same optimization techniques. The only difference here is the approximation of the second term of the problem (7.17).

In the previous stage we assumed that the distribution of the grey levels in the pixel could be represented by a single Gaussian with a mean value that corresponds to the mean value of the mixture of Gaussians and the standard deviation that corresponds to the standard deviation of the mixture of the Gaussians (7.15), (7.16). In this stage we assume that the grey levels distribution in each pixel is a Gaussian distribution with the mean value $\tilde{\mu}_j$ equal to the mean value of the most probable class and standard deviation equal to the standard deviation of the most probable class.

In other words, if

$$k_j = \arg\max_k \delta_{jk}$$

then

$$\tilde{\mu}_j = \mu_{k_j}, \tilde{\sigma}_j = \sigma_{k_j}.$$

As in the previous stage, first we do the image optimization

$$\begin{aligned} \mathbf{x}^{l+1} = \arg\max_{\mathbf{x}} \lambda_1 \left[\sum_i b_i \log \sum_j a_{ij} x_j - \sum_{i,j} x_j a_{ij} \right] + \\ + \sum_j \frac{(x_j - \tilde{\mu}_j)^2}{2\tilde{\sigma}_j^2}, \\ \text{s.t. } \forall j \quad x_j > 0. \end{aligned} \tag{7.25}$$

And, knowing the new image approximation, we do the segmentation of this image:

$$\begin{aligned} \delta^{l+1} = \arg\max_{\delta} \sum_j \log \sum_k \delta_{jk} \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j^{l+1} - \mu_k)^2}{2\sigma_k^2} \right) - \\ - \lambda_2 \|D\delta\|_2 \\ \text{s.t. } \forall j \quad \sum_k \delta_{jk} = 1; \ \forall j,k \quad \delta_{jk} \geq 0. \end{aligned} \tag{7.26}$$

After few iterations we can stop the process.

Require: $L_1, L_2, L_1 \leq L_2, \beta < 1, \lambda_1, \lambda_2, \{\mu_k\}, \{\sigma_k\}, A, \mathbf{b}$
$\mathbf{x} \leftarrow$ ones
$\delta \leftarrow$ equal probabilities
$l \leftarrow 0$ (iteration index)
while $l < L_1$ **do** ▷ SRS-II procedure, stage 1
$l \leftarrow l + 1$
Compute $\tilde{\mu_j}$, $\tilde{\sigma_j}$ according to (7.15), (7.16)
Compute λ_1^l according to (7.20) or σ_k^l according to (7.21).
$\mathbf{x}^{l+1} \leftarrow$ result of 20 iterations of L-BFGS optimization of (7.22) or (7.23) using $\tilde{\mu}_j$, $\tilde{\sigma}_j$ with fixed δ^l
$\delta^{l+1} \leftarrow$ result of 20 iterations of Frank-Wolfe optimization of (7.24) with fixed $\mathbf{x}^{l+1}$
end while
while $l < L_2$ **do** ▷ SRS-II procedure, stage 2
$l \leftarrow l + 1$
$\mathbf{x}^{l+1} \leftarrow$ result of 20 iterations of L-BFGS optimization of (7.25) with fixed δ^l
$\delta^{l+1} \leftarrow$ result of 20 iterations of Frank-Wolfe optimization of (7.26) with fixed $\mathbf{x}^{l+1}$
end while
return $\mathbf{x}^l$, δ^l

Figure 7.1: SRS-II algorithm.

7.4.3 Additional aspects of the algorithm

As we have mentioned, the regularization parameter λ_1^l or the class standard deviations σ_k^l change from iteration to iteration according to formulae (7.20) (7.21). In our algorithm we use the following values in these formulae: $C = 1000$, $\beta = 0.9$. The plot of the values $\frac{\lambda_1^l}{\lambda_1}$ and $\frac{\sigma_k^l}{\sigma_k}$ can be found in Figure 7.10.

Quite important question is selection of the starting point for each of the the optimization subproblems in each iteration. We select the previous iterates for both image and HMMFM as a starting point for both optimization problems. As for the initial guess for the overall algorithm, we choose the initial image equal to all ones, the initial HMMFM is chosen as equal probabilities for all the classes in all the pixels, but the initial guess should not affect the result of the process.

The algorithm listing can be found on figure 7.1.

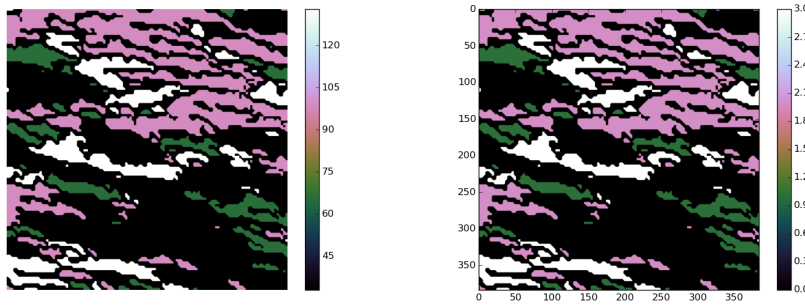

Figure 7.2: Ground truth of the experiment. Left - the phantom, right - the segmentation. The phantom is generated using the `AIR Tools` package.

7.5 Computational Results

To check the reconstruction properties of our algorithm we have done a series of experiments. We generated artificial phantoms of size 384×384 pixels with following mean values: [33, 66, 99, 133], see Figure 7.2.

The geometry for the problem was generated using the package `ASTRA` [18]. The advantage of this package is that instead of generating the explicit projection matrix A which corresponds to the geometry of the scanning device, the information about the geometry is stored. That allows to make matrix-vector multiplications on the GPU.

We generated two data sets, one with 172 projections and the other with 86 projections, both sets have $\lfloor \sqrt{2} \cdot 384 \rfloor = 543$ rays in each projection. The projections were produced using parallel-beam geometry. The projection angles are evenly distributed between 0 and 180 degrees in both data sets. Thus, the ratio between amount of data and amount of pixels is $\frac{\#data}{\#pixels} = 0.62$ for the first dataset and $\frac{\#data}{\#pixels} = 0.31$ for the second dataset meaning that both problems are underdetermined. Moreover, some of the rays do not hit the phantom at all because of the square geometry of the phantom. We assume the noise in the data is Poisson-distributed, which corresponds to an Emission Tomography problem. Each measurement corresponds to the integral of the emission coefficient along a

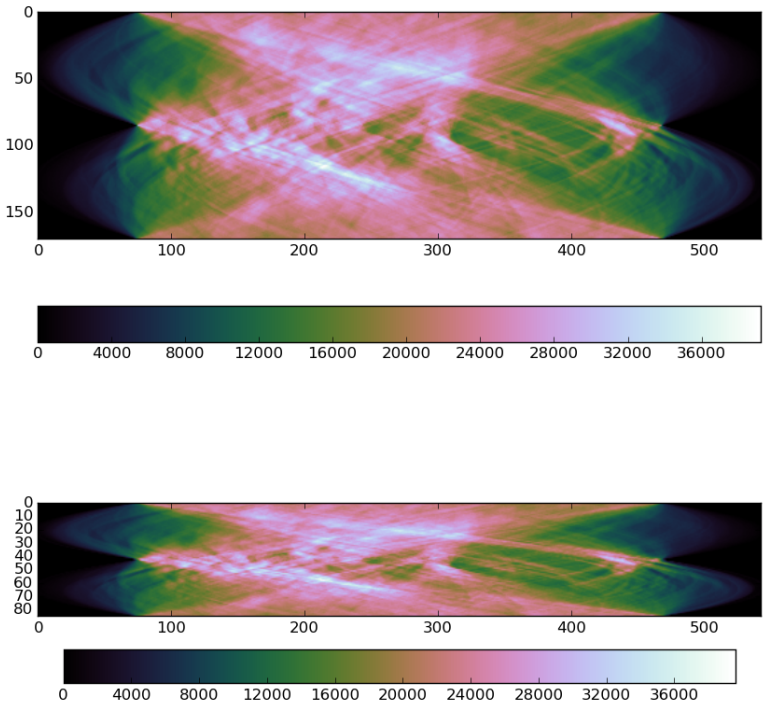

Figure 7.3: The sinograms of two experiments. Top: for 170 projections, bottom: for 85 projections. The noise level in the data is $\frac{\|\mathbf{b}-\mathbf{b}^*\|_2}{\|\mathbf{b}^*\|_2} = 0.0068$ in both cases, $\mathbf{b}^*$ is the data without noise, amount of rays in each of the projections is 543.

measurement line. We get the following quantity for the amount of noise:

$$\frac{\|\mathbf{b}-\mathbf{b}^*\|_2}{\|\mathbf{b}^*\|_2} = 0.0068,$$

for both data sets, where $\mathbf{b}^*$ is the data without noise.

The original phantom and the segmentation are shown in figure 7.2. The sinograms are shown in figure 7.3.

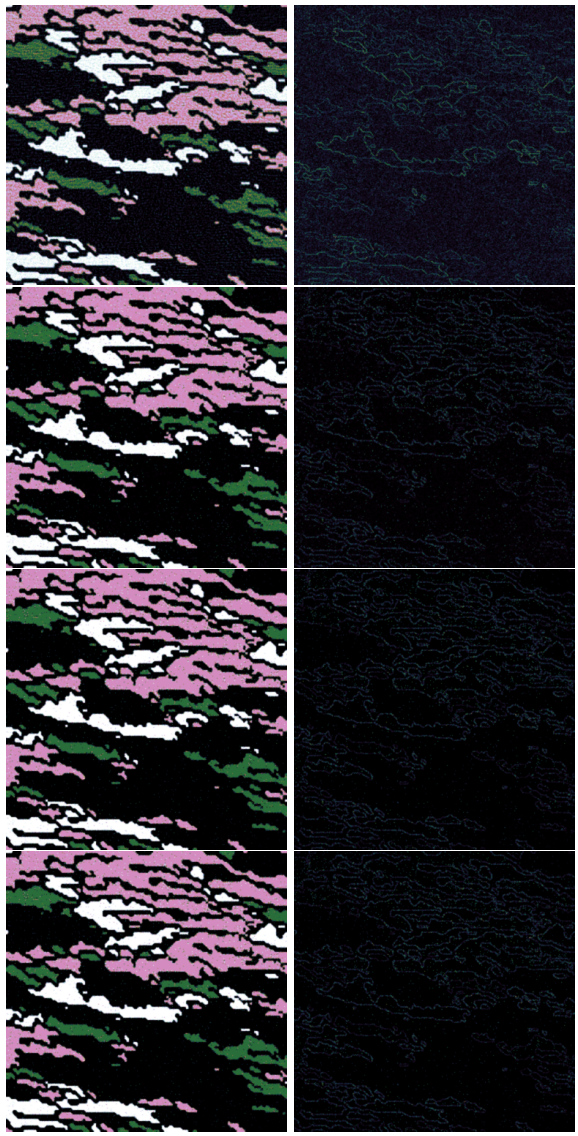

Figure 7.4: Left - reconstructed images, right - difference between current image and the ground truth $|\mathbf{x}^l - \mathbf{x}^*|$ (the $\mathbf{x}^*$ is a true image) for SRS-II algorithm applied to the test problem with 170 projections. The following iterations represented: 1, 33, 66, 100.

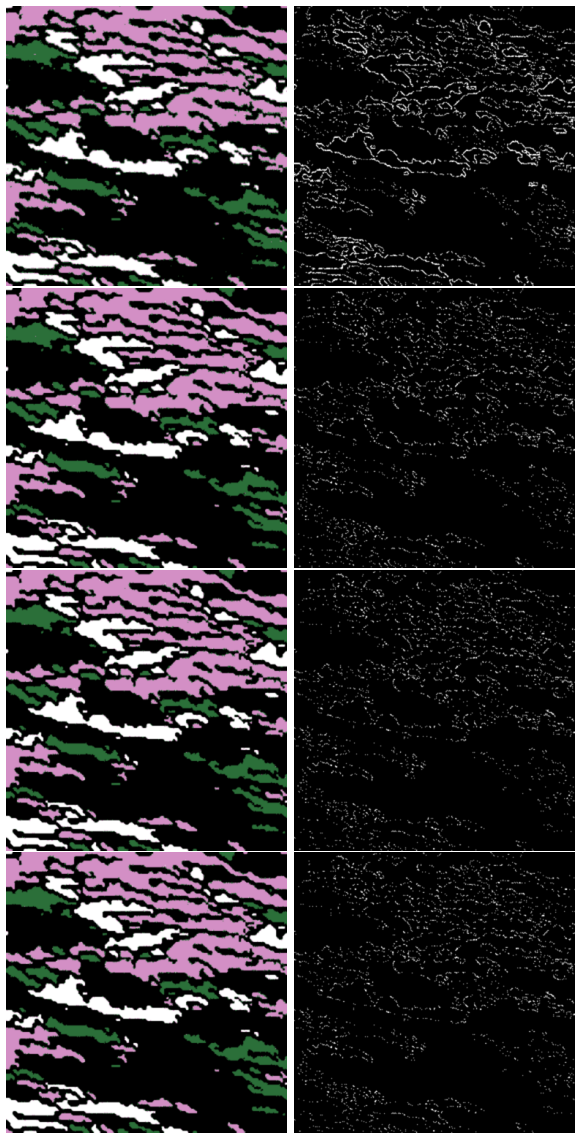

Figure 7.5: Left - computed segmentations, right - misclassified pixels (white) for SRS-II algorithm results computed from 170 projections. The following iterations are presented: 1, 33, 66, 100.

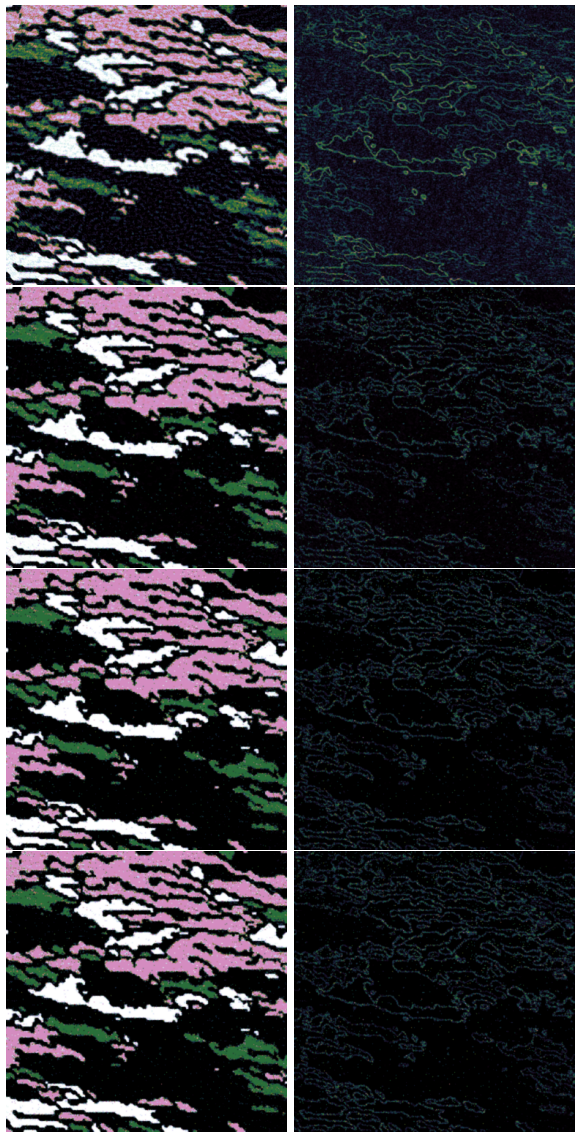

Figure 7.6: Left - reconstructed images, right - difference between current image and the ground truth $|\mathbf{x}^l - \mathbf{x}^*|$ (the $\mathbf{x}^*$ is a true image) for SRS-II algorithm applied to the test problem with 85 projections. The following iterations represented: 1, 33, 66, 100.

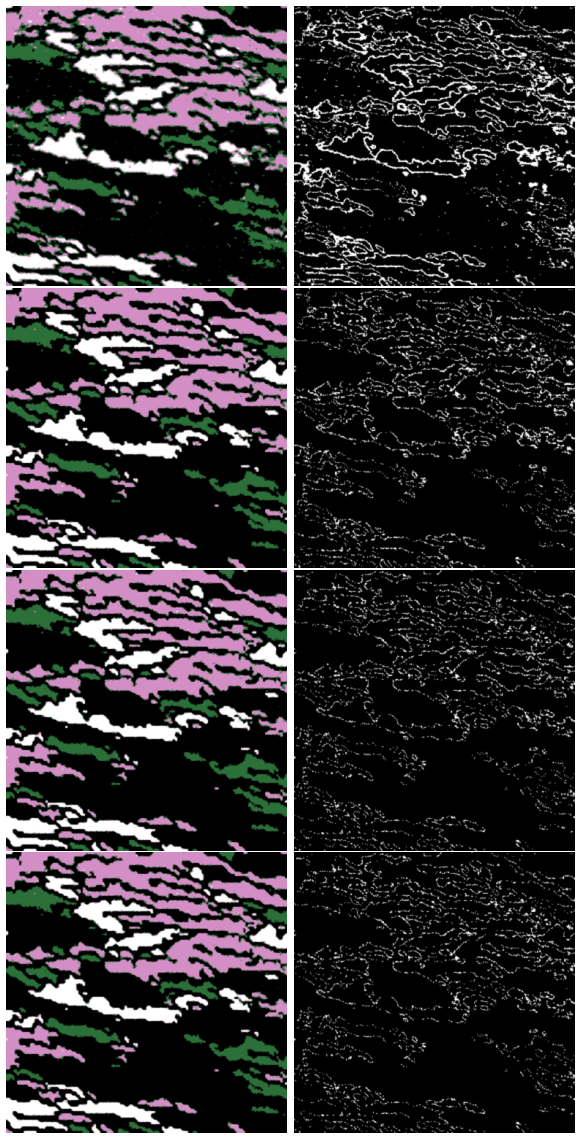

Figure 7.7: Left - computed segmentations, right - misclassified pixels (white) for SRS-II algorithm results computed from 85 projections. The following iterations are presented: 1, 33, 66, 100.

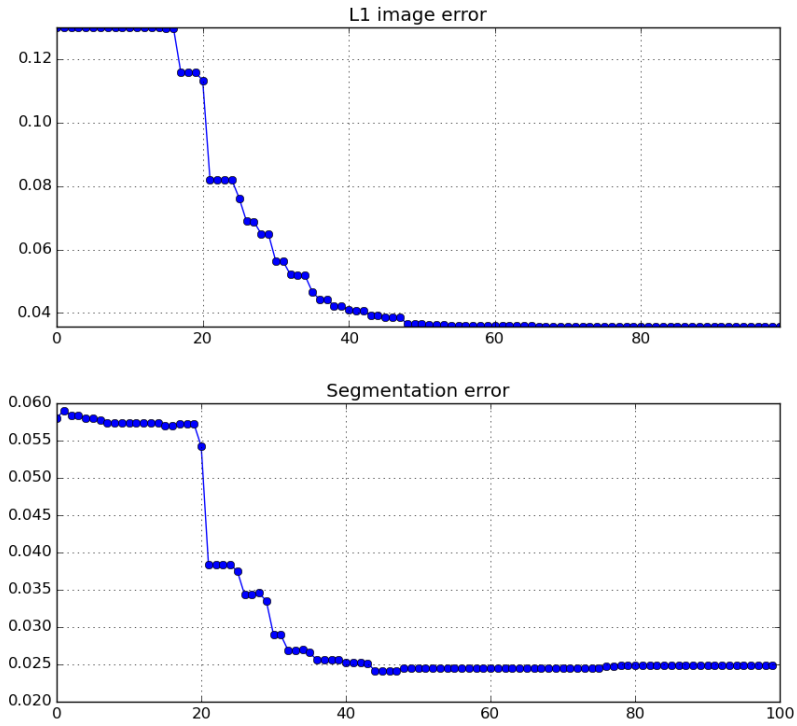

Figure 7.8: Error histories for SRS-II problem with 170 projections. Top: error history for l_1 image error $\frac{\|\mathbf{x}-\mathbf{x}^*\|_1}{\|\mathbf{x}^*\|_1}$. Bottom: segmentation error - the fraction of misclassified pixels. The $\mathbf{x}^*$ is a true image.

In the numerical experiments we expect to see that the use of the segmentation in the reconstruction process, jointly updating the image and the segmentation, will improve both the image and the segmentation. In the beginning of the iterations the regularization parameter λ_1^l is quite big meaning that the class fitting term and the regularization term have very little influence on the result of the optimization process. The fact that we get a better reconstruction during the iterations indicates that using the segmentation in the reconstruction process we are able to produce a better reconstruction. This is due to the "feedbck effect" between the segmentation and the reconstruction.

For the reconstruction process the following parameters were chosen: for the problem with 170 projections: $\lambda_1 = 250, \lambda_2 = 0.8$, for the problem with 85

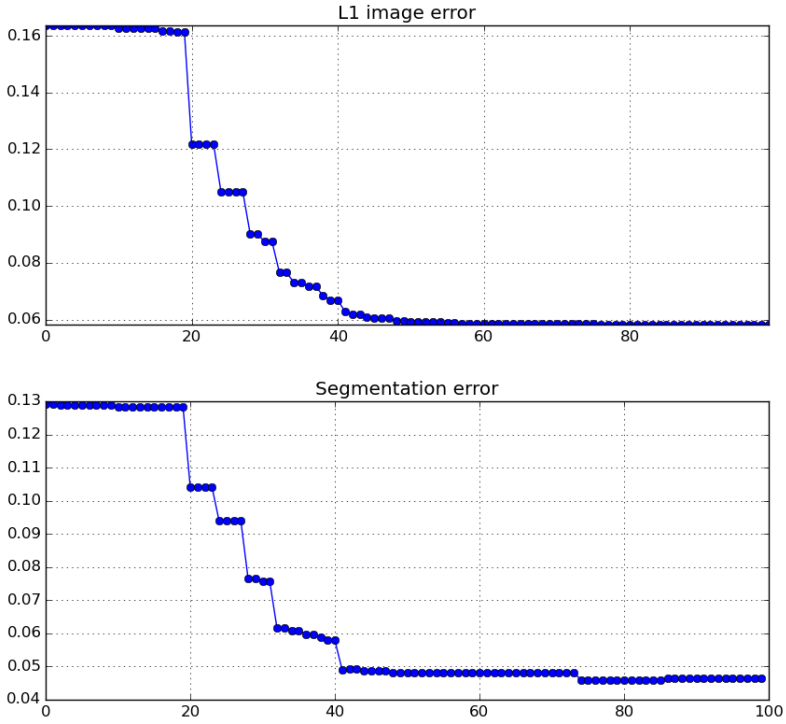

Figure 7.9: Error histories for SRS-II problem with 85 projections. Top: error history for l_1 image error errors $\frac{\|\mathbf{x}-\mathbf{x}^*\|_1}{\|x^*\|_1}$. Bottom: segmentation error - the fraction of misclassified pixels. The $\mathbf{x}^*$ is a true image.

projections: $\lambda_1 = 800, \lambda_2 = 1.0$. These parameters were obtained using trial-and-error and produced the best possible result.

We show the evolution of the reconstructions of the image and segmentations as well as the differences between the reconstruction of the phantom and misclassified pixels. Figures 7.4 and 7.5 show results for 170 projections and figures 7.6, 7.7 show results for 85 projections. The amount of misclassified pixels decreases with iterations, the quality of reconstruction increases over the iterations. We also see that with iterations the precision of location of the edges increases. From the figures one can conclude that the majority of misclassifications in the segmentation and errors in the reconstruction occur on the edges between the classes. This is easy to explain: the lower is the amount of photons - the less reliable is

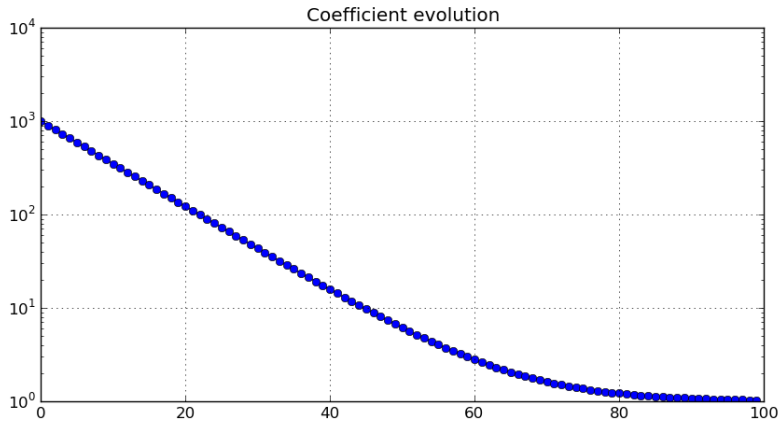

Figure 7.10: Evolution of the coefficients.

the data and the less precise the possible reconstruction could be. As a result, the position of the edge is less predictable, and this leads to the misclassifications on the edges. The error histories are presented in figure 7.8 for 170 projections and figure 7.9 for 85 projections. As a measure for the image error we used the relative l_1 error:

$$\frac{\|\mathbf{x}^l - \mathbf{x}^*\|_1}{\|\mathbf{x}^*\|_1},$$

where $\mathbf{x}^*$ is the true image We have chosen this error because it penalizes small deviations and big deviations equally and, because of that, corresponds better to visual inspection. As an error measure for the segmentation we have chosen the fraction of misclassified pixels:

$$\frac{\sum_j I(s_j \neq s_j^*)}{\#pixels},$$

where I is a indicator function that is 0 when the argument is false and 1 when the argument is true; s^* is a true segmentation.

We can see that both reconstructions and segmentations improve along the iterations. In the same time, the reconstructions that were obtained with these methods are piecewise constant and the edges are sharp, staircasing artifacts are minimal. Besides, the noise is suppressed by the regularization.

We have also compared the performance of the different flavours of the algorithm

(Table 7.2). First experiment was done with the SRS-II algorithm with 2-norm data fitting term

$$\|A\mathbf{x} - \mathbf{b}\|_2$$

with the data with $\frac{\#data}{\#pixels} = 0.3$ ratio between amount of data and amount of pixels (85 projections) and with 0.01 relative Gaussian noise. The reconstruction algorithm used standard deviations σ_k that were equal for all the classes and were identical and equal to 0.001. The regularization parameters that were used for this experiment are: $\lambda_1 = 0.9, \lambda_2 = 1.0$. Another experiment used Poisson data fitting term (the definition for the data fitting term for the Poisson noise is given in (7.17)), the amount of data over the amount of pixels was $\frac{\#data}{\#pixels} = 0.3$ (85 projections), relative amount of Poisson noise was 0.0068, the standard deviations of classes were set to be equal to $\sigma_k = 0.001\mu_k$. The regularization parameters that were used for all the other problems are: $\lambda_1 = 2000.0, \lambda_2 = 0.8$. The difference between the third experiment and the second is only in the standard deviations of the classes. We need this comparison to be able to tell if the choice of the standard deviations may compensate for the features of the Poisson Noise: it tends to have bigger errors in places, where the intensity is higher. Here we used the standard deviations that are equal to 0.01. For comparison we added one more numerical experiment with more data: the ratio between amount of data and amount of pixels $\frac{\#data}{\#pixels} = 0.6$ (170 projections).

From this table one can make the following conclusion: though it may seem logical to use the standard deviation of the classes proportional to the mean values of the classes - this changes the results insignificantly and does not improve them. The problems with more data produce better results with the same regularization parameters. This is logical and predictable. Also, the algorithm is good for the reconstruction of the image from the data with Gaussian Noise too.

		Gaussian $\sigma_k = 0.001$, $\frac{\#data}{\#pixels} = 0.3$	Poisson $\sigma_k = 0.001\mu_k$, $\frac{\#data}{\#pixels} = 0.3$	Poisson $=\sigma_k = 0.001$, $\frac{\#data}{\#pixels} = 0.3$	Poisson $\sigma_k = 0.001$, $\frac{\#data}{\#pixels} = 0.6$
l_1 image error	σ_k reduction	0.06	0.07	0.061	0.035
	λ_1 reduction	-	0.077	0.08	0.048
segm. error	σ_k reduction	0.052	0.062	0.056	0.023
	λ_1 reduction	-	0.085	0.077	0.037

Table 7.2: The table of comparison of different reconstruction approaches.

7.6 Conclusion

We have developed an algorithm for Poisson noise in the data. It is suited for large-scale-problems as we do not store the matrix A explicitly. The algorithm performs well on artificial test problems. The priors σ_k for the classes can be either fixed, or proportional to μ_k, but the computed results are almost the same. The solutions that are generated by this algorithm have sharp edges.

7.7 Appendix: Explanation of Modified Standard Deviation

In the expression (7.23) and (7.22) we use the term σ_k^l and λ_1^l to gradually shrink the searching range of the image, starting with big standard deviations σ_k^l or high λ_1^l values that correspond to significantly underregularized problems. Along the iterations we reduce values λ_1^l and σ_k^l to the values λ_1 and σ_k and, by that we gradually increase the regularization of the problem.

To show the importance of this modification we will make the thought experiment: consider the problem (7.19), where $\tilde{\sigma}$ and $\tilde{\mu}$ are computed using the formulae (7.15), (7.16) two specific realization of δ. As an information about the classes we will take two classes

$$\mu_0 = 1,\ \mu_1 = 2,$$

$$\sigma_0 = 0.1,\ \sigma_1 = 0.1.$$

As an example of different HMMFM realizations we will take one with all equal probabilities:

$$\forall j, k\ \delta_{jk} = 1/2,$$

and another realization will have

$$\forall j\ \delta_{j0} = 1,\ \delta_{j1} = 0.$$

In both cases it is easy to estimate parameters $\tilde{\mu}_j$ and $\tilde{\sigma}_j$ for each pixels: in the first case the approximate parameters for each pixel are

$$\tilde{\mu}_j = 0.5,\ \tilde{\sigma}_j = \sqrt{0.26} \approx 0.5.$$

in the second case the approximate parameters for each pixel are:

$$\tilde{\mu}_j = 1.0,\ \tilde{\sigma}_j = 0.1.$$

In the first case - the standard deviation is large enough - meaning that any combination of grey levels of the pixels in the range between 0 and 1 is acceptable as far as it satisfies the data fitting term.

In the second case the standard deviation is small, meaning that the acceptable range of the grey levels of the pixels is in the range from 0.9 to 1.1. Hence, the data in this case may be fitted significantly worse due to these limits - and that means that the image that we can get in the end of the optimization process is overregularized.

In case after this image optimization step we will try to optimize the HMMFM - in the first case we will get HMMFM that corresponds to the image that fits data quite well. In the second case - we will get a HMMFM that corresponds to the overregularized image - meaning that HMMFM has very small chance to change.

That automatically means that different initializations of the HMMFM may lead to completely different results due to this fact. Another problem that is a consequence of this effect is that the algorithm without modifications tends to stuck in the local minimum. We would like to get rid of both these problem that may spoil the results of the algorithm.

Consider now the modified standard deviations:

$$\sigma_k^l = \sigma_k(1 + C\beta^l).$$

Consider also $\sigma_k = 0.001$, $C = 1000$, $\beta = 0.9$. In the second case the result of the 7.15 and 7.16 with $l = 0$ will be

$$\tilde{\mu}_j = 1.0, \tilde{\sigma}_j = 1.0$$

that will mean that the image in the beginning (when l is small) can change significantly. On the other hand, consider $l = 100$. In this case the value of the expression 7.15 will not change, but the value of 7.16 will be

$$\sigma_j \approx 0.001$$

meaning that in the end the result is regularized significantly by HMMFM δ. This approach is related to the Simulated Annealing approach.

The same effect will take place when, instead of σ_k we shrink the λ_1 over the iterations:

$$\lambda_1^l = \lambda_1(1 + C\beta^l).$$

Although, in the first case the form of the class prior also changes, while in this case only the level of regularization changes. From our experience, it is better to change σ_k over the iterations, but shrinking λ_1 it is also possible to get a good result.

Bibliography

[1] Kyongtae T Bae et al. "Automatic Segmentation of Liver Structure in CT Images". In: *Medical Physics* 20.1 (1993), pp. 71–78.

[2] Dimitri P. Bertsekas. *Nonlinear Programming, 2. Ed.* Athena Scientific, Belmont, MA, 1999.

[3] Junguo Bian et al. "Evaluation of Sparse-View Reconstruction from Flat-Panel-Detector Cone-Beam CT". In: *Physics in Medicine and Biology* 55.22 (2010), p. 6575.

[4] Yuri Boykov and Gareth Funka-Lea. "Graph Cuts and Efficient ND Image Segmentation". In: *Int. J. Comp. Vis.* 70.2 (2006), pp. 109–131.

[5] Yuri Boykov, Olga Veksler, and Ramin Zabih. "Fast Approximate Energy Minimization via Graph Cuts". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 23.11 (2001), pp. 1222–1239.

[6] Yuri Y Boykov and M-P Jolly. "Interactive Graph Cuts for Optimal Boundary and Region Segmentation of Objects in ND Images". In: *Proceedings of the Eighth IEEE International Conference on Computer Vision.* Vol. 1. IEEE. 2001, pp. 105–112.

[7] Ronald Newbold Bracewell and ACf Riddle. "Inversion of Fan-Beam Scans in Radio Astronomy". In: *The Astrophysical Journal* 150 (1967), p. 427.

[8] Richard H Byrd et al. "A limited memory algorithm for bound constrained optimization". In: *SIAM Journal on Scientific Computing* 16.5 (1995), pp. 1190–1208.

[9] Tony F Chan, Gene H Golub, and Pep Mulet. "A Nonlinear Primal-Dual Method for Total Variation-Based Image Restoration". In: *SIAM Journal on Scientific Computing* 20.6 (1999), pp. 1964–1977.

[10] Tommy Elfving, Per Christian Hansen, and Touraj Nikazad. "Semi-convergence properties of Kaczmarz's method". In: *Inverse Problems* 30.5 (2014), p. 055007.

[11] Per Christian Hansen and Maria Saxild-Hansen. "AIR tools—a MATLAB package of algebraic iterative reconstruction methods". In: *Journal of Computational and Applied Mathematics* 236.8 (2012), pp. 2167–2178.

[12] S Kaczmarz. "Approximate solution of systems of linear equations". In: *International Journal of Control* 57.6 (1993), pp. 1269–1271.

[13] Avinash C.. Kak and Malcolm Slaney. *Principles of computerized tomographic imaging*. Society for Industrial and Applied Mathematics, 2001.

[14] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. "Snakes: Active contour models". In: *International journal of computer vision* 1.4 (1988), pp. 321–331.

[15] Jose L Marroquin, Edgar Arce Santana, and Salvador Botello. "Hidden Markov measure field models for image segmentation". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25.11 (2003), pp. 1380–1387.

[16] Carl D Meyer. *Matrix analysis and Applied linear algebra*. SIAM, 2000.

[17] Stanley Osher and Nikos Paragios. *Geometric level set methods in imaging, vision, and graphics*. Springer Science & Business Media, 2003.

[18] Willem Jan Palenstijn, K Joost Batenburg, and Jan Sijbers. "The ASTRA Tomography Toolbox". In: *13th International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE*. Vol. 2013. 2013.

[19] Mikhail Romanov et al. "Simultaneous Tomographic Reconstruction and Segmentation with Class Priors". In: *submitted to Inverse Problems in Science and Engineering* (2015).

[20] James Albert Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press, 1999.

[21] Emil Y Sidky, Chien-Min Kao, and Xiaochuan Pan. "Accurate Image Reconstruction from Few-Views and Limited-Angle Data in Divergent-Beam CT". In: *Journal of X-ray Science and Technology* 14.2 (2006), pp. 119–139.

[22] Dominique Van de Sompel and Michael Brady. "Simultaneous reconstruction and segmentation algorithm for positron emission tomography and transmission tomography." In: *ISBI*. 2008, pp. 1035–1038.

[23] David Strong and Tony Chan. "Edge-Preserving and Scale-Dependent Properties of Total Variation Regularization". In: *Inverse Problems* 19.6 (2003), S165.

Chapter 8

A Parameter Choice Method for Simultaneous Reconstruction and Segmentation

The problem of finding good regularization parameters for the reconstruction problems without knowledge of the ground truth is a non-trivial task. We overview the existing parameter-choice methods and present the modified L-curves approach for a good regularization parameters selection that is suited for our Simultaneous Reconstruction and Segmentation method. We verify the validity of this approach with numerical experiments based on reconstructions of artificial phantoms from noisy data, and the problems in our numerical experiments are underdetermined.

8.1 Introduction

In this work we consider tomographic reconstruction of an object's interior from X-ray transmission projections. The geometry of the projections is parallel-beam geometry. The amount of data is less than the amount of the pixels in image - thus, the problem is underdetermined. Also, the data is not perfect and contains some amount of noise.

To deal with this problem we formulate a minimization problem that contains three terms:

$$\lambda_1 \mathcal{D}(\mathbf{x}), \mathcal{C}(\mathbf{x}, \delta), \lambda_2 \mathcal{R}(\delta), \tag{8.1}$$

where $\mathbf{x}$ is the image, δ is a Hidden Markov Measure Field Model (will be explained later), $\mathcal{D}$ is a data fitting term, $\mathcal{C}$ is a class fitting term, $\mathcal{R}$ is a segmentation regularization term. To balance these terms we need two regularization parameters: λ_1, λ_2. In order to make the best reconstruction, we need to find the best regularization parameters. In this work we consider an approach that may help to find good regularization parameters.

The parameter selection is an important problem for the reconstruction process. In case the ground truth of the reconstruction is known, then the process of selection of the optimal parameters is trivial: the set of parameters are checked and after that the parameters that leads to the smallest errors are selected.

In real-world problems this task may not be trivial as the ground truth is not known. In this case the choice of the best parameters for the reconstruction algorithm is hard, and we need to use some heuristics to understand how good or how bad is the resulting reconstruction. Also, the fact that usage of these heuristics does not perform well in all cases should be accepted. In addition, each of the heuristics has its own limitations. In some cases one of the heuristics will perform very well, in other cases it will fail.

In this work we will make our proposal for a good heuristic that will lead to a good choice of the regularization parameters for the Simultaneous Reconstruction and Segmentation Algorithm (SRS) [15].

The main idea of the SRS algorithm is to compute a reconstruction with a segmentation (i.e. the different materials locations) in a joint fasion. The information about the materials is given beforehand. We use this information to make the reconstructions and segmentation more precise than what is possible to do in a classical, non-simultaneous way. The SRS algorithm is a variational algorithm,

the minimization problem has three terms and two regularization parameters.

There is a bunch of well-known techniques to choose the regularization parameter. One approach to solve this problem is the discrepancy principle (DP) [11][1][17]. The main idea of this method is that the discrepancy between the measured data and the data predicted by the reconstructed image should be comparable to the standard deviation of the noise. In case the discrepancy is bigger then the standard deviation of noise, that means that the solution may be computed better. In case it is smaller, then there is high risk of overfitting the noise. This approach is applicable to problems with Gaussian noise. In case the noise is different from Gaussian, then the estimation of discrepancy may be a tough problem. Besides, the amount of noise needs to be known. This also causes some limitations.

A modification of the previous method that generalizes the approach and makes it possible to use it without knowing the exact level of the noise is called the L-curve method [3][8][6]. The main idea of this approach is that a good regularization parameter represents a good balance between the data fit and the regularization. This good compromise is usually seen on the plot of the regularization term plotted versus the data fitting term and represented by a corner of the graph.

Another important method is Generalized Cross-Validation (GCV) [2][13][10]. The main idea of cross validation is that reconstruction of the image is done from incomplete data, where one of the instances is left out, and then we compute the estimated value of the left out instance from the image reconstruction. In principle we do this procedure for each of the data instances and the regularization parameter is chosen such that the prediction errors are minimized. GCV is another method for doing this. Instead of doing the reconstruction for leaving out each of the data points, one may do only one reconstruction for each of regularization parameters and then to compute the residual norm normalized by a factor that takes into account the reduction of the amount of degrees of freedom the measurement geometry. The objective is to find the regularization parameters that minimize this normalized sum of the errors. This method is one of the most efficient ones, but it has a significant drawback: defining and computing the denominator for the GCV function is a difficult problem, except for Tikhonov Regularization [16] and Truncated SVD [7].

One more important method of parameter selection is called Normalized Cumulative Periodogram (NCP) [12]. The idea of this method lies in the field of spectral analysis. In this approach we analyze the difference between the real data and the data predicted by the image, referred to as residual. If the solution is un-

derregularized, the high frequencies are more represented in the residual. In case the solution is overregularized, the low frequencies will be more represented in the residual. The optimal regularization parameter, from the point of view of the NCP, represents an equal amount of high and low frequencies. The shortcoming of this approach that to use this criterion, we need to know or to be able to compute the spectrum of the problem.

More information about the regularization parameter choice methods can be found in the books [5], [4].

To sum up, we have four candidate strategies to find the optimal regularization parameters: DP, L-curve, GCV, NCP. The use of GCV is not possible as we do not know how to define the normalization coefficients in the error function. For the DP we need to know the norm of the noise that is not always known. As for the other methods, it makes sense to try them.

8.2 Brief Description Of the SRS Method

In this work we reconstruct the object $\mathbf{x}$ from the projections $\mathbf{b}$. The underlying problem has the form:

$$\mathbf{b} = A\mathbf{x} + \epsilon, \tag{8.2}$$

where ϵ is the vector of noise in each datum. In this work we assume that the noise is Gaussian and is independent in all different measurements, has zero mean and the same standard deviation in each datum. The matrix A corresponds to the projections geometry. The element of this matrix a_{ij} is the length of the ray that corresponds to the measurement i inside of the pixel j.

We want to make a reconstruction of the object's attenuation coefficients $\mathbf{x}$ from the data $\mathbf{b}$ with the assumption that the object consists of several materials, each of these materials is characterized by the mean value μ_k of the attenuation coefficient and the standard deviation σ_k of the attenuation coefficient and with Gaussian distribution of the attenuation coefficient. Thus, the attenuation coefficient has the following distribution given that it belongs to class k:

$$p(x_j|k) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right). \tag{8.3}$$

The algorithm that we use for the tomographic reconstruction of the image is the Simultaneous Reconstruction and Segmentation (SRS) algorithm. This algorithm

uses the concept of a Hidden Markov Measure Field Model (HMMFM) as the model for the segmentation. This model assigns to each of the pixels j the probability δ_{jk} to belong to each of the possible classes k. As these values are probabilities, they sum to 1:

$$\forall j \quad \sum_k \delta_{jk} = 1. \tag{8.4}$$

Also, the probabilities can not be negative:

$$\forall j,k \quad \delta_{jk} \geq 0. \tag{8.5}$$

The problem that is solved is formulated in the following way:

$$\begin{aligned} \min_{\mathbf{x},\delta} \quad & \lambda_1 \|A\mathbf{x} - \mathbf{b}\|_2^2 - \\ & \sum_j \log \sum_k \frac{\delta_{jk}}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right) + \\ & \lambda_2 \sum_j \sum_{j' \in N_j} \sum_k (\delta_{jk} - \delta_{j'k})^2 \\ \text{s.t.} \quad & \forall j \ \sum_k \delta_{jk} = 1, \\ & \forall j,k \ \delta_{jk} \geq 0. \end{aligned} \tag{8.6}$$

Here x_j is the value of the attenuation coefficient inside pixel j, λ_1 and λ_2 are the regularization parameters that we would like to find, N_j is the set of neighbours of pixel j. This problem is non-convex because of the second term. In this approach we use μ_k and σ_k as a prior knowledge about the object under the reconstruction.

To solve this problem an iterative two-step approach is used. In the first stage we find the new image iterate solving the problem

$$\begin{aligned} \mathbf{x}^{l+1} = \arg\min_{\mathbf{x}} \quad & \lambda_1 \|A\mathbf{x} - \mathbf{b}\|_2^2 + \\ & \sum_j \frac{(x_j - \hat{\mu}_j)}{2\hat{\sigma}_j}, \end{aligned} \tag{8.7}$$

where $\mathbf{x}^{l+1}$ is the $l+1$ iterate for $\mathbf{x}$, the values $\hat{\mu}_j$ and $\hat{\sigma}_j$ are the expected value of the pixel j attenuation coefficient and the standard deviation of the pixel j attenuation coefficient:

$$\tilde{\mu}_j = \sum_k \delta_{jk} \mu_k \tag{8.8}$$

$$\tilde{\sigma}_j^2 = \sum_k \delta_{jk}(\mu_k^2 + \sigma_k^2) - \hat{\mu}_j^2. \tag{8.9}$$

The substitutions (8.7), (8.8), (8.9) of the original image optimization problem that is part of the problem (8.6) makes it convex and easy to solve.

After the image has been updated, we update the HMMFM solving the problem:

$$\begin{aligned} \delta^{l+1} = \arg\min_{\delta} \quad & -\sum_j \log \sum_k \frac{\delta_{jk}}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j^{l+1} - \mu_k)^2}{2\sigma_k^2}\right) + \\ & + \lambda_2(\delta_{jk} - \delta_{j'k})^2 \\ s.t.\ \forall j \quad & \sum_k \delta_{jk} = 1, \\ \forall j,k \quad & \delta_{jk} \geq 0. \end{aligned} \tag{8.10}$$

After the HMMFM is updated using (8.10), the image is recomputed again according to (8.7) and so forth. After the convergence of this process it is possible to add one more stage of making the image more sharp. This enhancement deals with the fact that we do not solve the original problem (8.6), but a simplified version of it. From our experience, this makes sense only in case the iterative process described above converges to a good approximation of the solution. Thus, we do not consider this enhancement here as the optimal parameters for this procedure should also be optimal for the procedure above.

To sum up, our aim is to find the regularization parameters for the reconstruction procedure, described by (8.7), (8.10). For more details and numerical examples, see [14].

8.3 Parameter Selection Algorithm

To find a good parameter selection algorithm we have modified the L-curve criterion.

The original L-curve criterion is formulated as follows: suppose we have the data fitting term $\mathcal{D}(\mathbf{x})$ (in our case this is $\|A\mathbf{x} - \mathbf{b}\|_2$) and the regularization term $\mathcal{R}(\mathbf{x})$, where both terms are convex. In this case the whole problem may be

formulated as:

$$\min_{\mathbf{x}} \lambda \mathcal{D}(\mathbf{x}) + \mathcal{R}(\mathbf{x}). \tag{8.11}$$

Then the L-curve is defined as a log-log plot of $\mathcal{D}(\mathbf{x})$ against $\mathcal{R}(\mathbf{x})$ for different regularization parameters λ. As an example, in case the reconstruction problem is formulated as

$$\min_{\mathbf{x}} \lambda \|A\mathbf{x} - \mathbf{b}\|_2^2 + \sum_j \|D_j \mathbf{x}\|_2, \tag{8.12}$$

where D_j is a discrete approximation of the gradient length in the pixel j, then the L-curve is the plot of $\|A\mathbf{x} - \mathbf{b}\|_2$ against $\sum_j \|D_j\mathbf{x}\|_2$ as a function of the regularization parameters. The best regularization parameter is usually supposed to be in the corner of the L-curve that represents a balance between the data fitting term and the regularization term. In case the regularization parameter deviates to the larger side, the more emphasis is put on the data fitting term and the more is the overfitting of the noise. This corresponds to a significant growth of the regularization term $\mathcal{R}(\mathbf{x})$ and insignificant improvement of the data fitting term $\mathcal{D}(\mathbf{x})$. In case the regularization parameter is lower then the optimal one, less emphasis is put on the data fitting term and, thus, the result is overregularized. This situation corresponds to insignificant improvement of the regularization term $\mathcal{R}(\mathbf{x})$ and significant growth of the data fitting term $\mathcal{D}(\mathbf{x})$.

Our problem has a nice variational formulation and, thus, the L-curve is a good candidate for the parameter selection approach. On the other hand our problem is a non-convex problem (see (8.6)) and this may cause some difficulties during analysis of the L-curve. To deal with it, we need to make some modifications to the L-curve approach. Also, we have two regularization parameters. This makes the problem significantly more complicated. We also have to find a way to deal with this.

First, let us consider the task of finding a good regularization parameter λ_1 for our reconstruction problem, when λ_2 is fixed.

The λ_1 regularization parameter sets the balance in Equation (8.10) between the data fitting term

$$\mathcal{D}(\mathbf{x})\|A\mathbf{x}(\lambda_1, \lambda_2) - \mathbf{b}\|_2^2 \tag{8.13}$$

and the class fitting term

$$\mathcal{C}(\mathbf{x}, \delta) - \sum_j \log \sum_k \frac{\delta_{jk}}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x_j(\lambda_1, \lambda_2) - \mu_k)^2}{2\sigma_k^2}\right). \tag{8.14}$$

Note that here the solution $\mathbf{x}(\lambda_1, \lambda_2)$ depends on the regularization parameters λ_1 and λ_2. The term (8.14) is the probability density value of the grey value

x_j given the class probabilities $\delta_{j,k}$ for all k. The L-curve in this case should be represented by terms (8.13) and (8.14) values plotted against each other. We suggest to substitute the second terms' values with the values computed using the following approximation:

$$\hat{\mathcal{C}}(\mathbf{x}, \delta) \sum_j \min_k \frac{(x_j - \mu_k)^2}{2\sigma_k^2}, \tag{8.15}$$

that corresponds to the value (8.14), where the actual values δ_{jk} are substituted by 1 for the class with highest probability density in the point x_j and 0 for all the other classes. The smaller the term (8.15), the closer the grey values of the pixels are to the means of the classes in terms of differences normalized by σ_k^{-1}. To use this approach is quite logical as the smaller is the distance of grey levels of pixels to the means of the classes, the larger is the influence of the class fitting term is. The bigger the distance the more noise in the data is overfitted.

Now, let us consider the problem of finding of good regularization parameter λ_2.

The parameter λ_2 corresponds to the balance between the last two terms in Equation (8.10). The first of the terms is the class fitting term (8.14). The second term is the HMMFM regularization term:

$$\mathcal{R}(\delta) = \sum_j \sum_k \sum_{j' \in N_j} (\delta_{jk} - \delta_{j'k})^2. \tag{8.16}$$

As before, the second term (8.14) is too complex to be analyzed. Instead of it, we will use the data fitting term (8.15) value for comparison.

We will present the results of this modified L-curve approach in the next section.

8.4 Results

To test our idea for choosing the regularization parameter we used two test problems. For both of these problems the artificial phantom was used as images $\mathbf{x}$. The first one was created using the standard `MATLAB` command `phantom`. Another one was the binary phantom from the `AIR Tools` package [9] and was created using the command `phantomgallery`. In both cases the phantom consists of 128×128 pixels. The phantoms are shown on Figure 8.1.

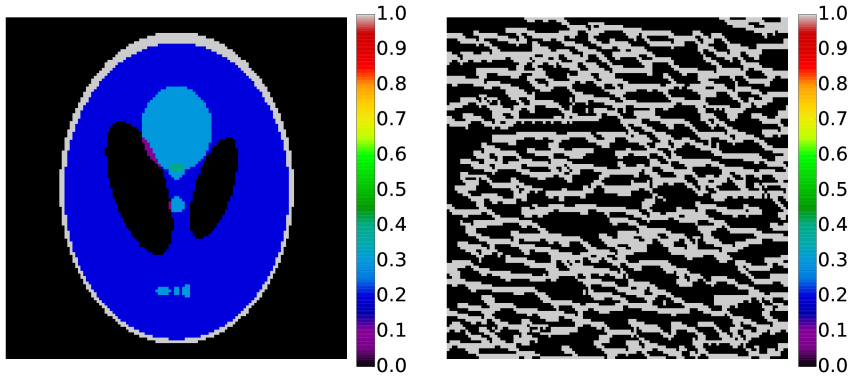

Figure 8.1: The phantoms that are used in the experiments. Left: Shepp-Logan phantom, generated with `MATLAB` command `phantom`, right: binary phantom, generated with command `phantomgallery` from package `AIR Tools`.

The first phantom has several typical attenuation coefficients: 0.0, 0.1, 0.2, 0.3, 0.4 and 1.0. The second phantom consists of only two classes of attenuation coefficients: 0.0 and 1.0.

The geometry of the projections for both problems was selected to have parallel-beam projections. The number of projections is 59, there are $\lfloor \sqrt{2} \cdot 128 \rfloor = 181$ rays in each of the projections. Thus, the ratio between amount of data and amount of pixels is equal to $\frac{\#data}{\#pixels} = 0.6$. Thus, all the problems are underdetermined. The projection matrix A was generated using the package `AIR Tools` using the function `paralleltomo`.

For both problems the data was computed according to the rule

$$\mathbf{b} = A\mathbf{x} + \varepsilon, \tag{8.17}$$

where for both problems ε is the different for each problem vector of independent Gaussian noise. The amount of this noise for both problems was set to be

$$\frac{\|\varepsilon\|_2}{\|b\|_2} = 0.01.$$

We have computed the solutions for a large set of regularization parameters. For each of the pairs of regularization parameters the relative image error was

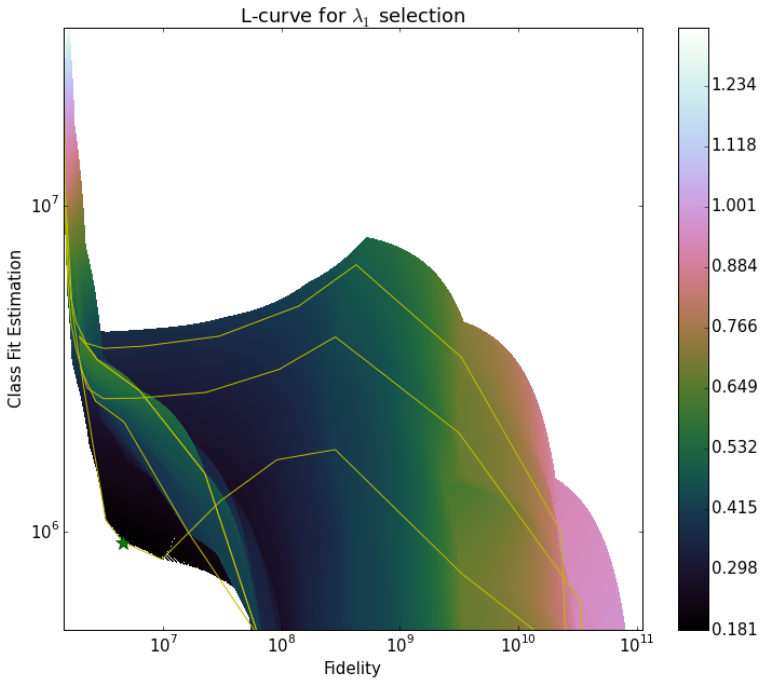

Figure 8.2: The modified L-curves plot for the Binary phantom for λ_1 parameter selection. The λ_2 parameter is constant along the yellow lines. The optimal pair of λ_1 and λ_2 parameters is denoted by a green star. The colour represents the relative image error in this point: $\frac{\|x-x^*\|_2^2}{\|x^*\|_2^2}$, where x^* is the ground truth, x is the reconstructed image.

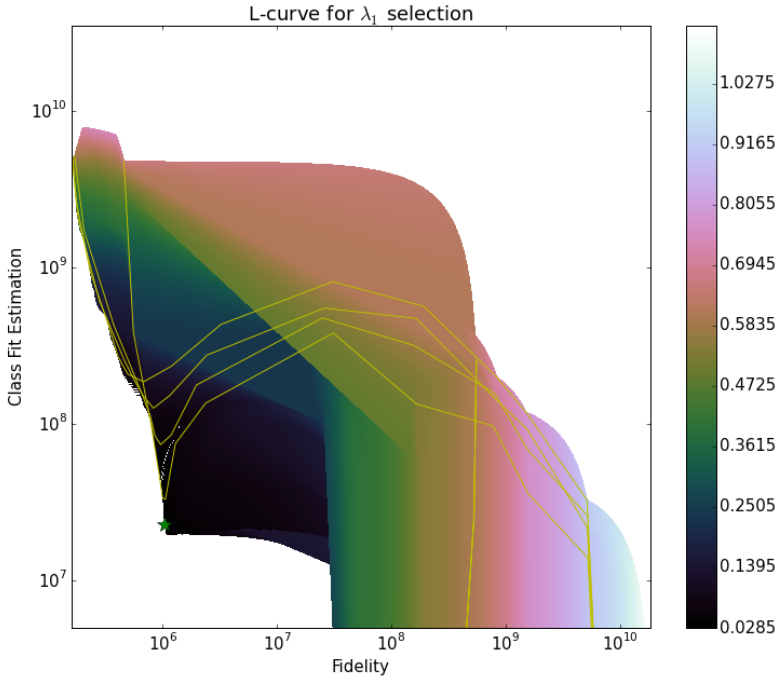

Figure 8.3: The modified L-curves plot for the Shepp-Logan phantom for λ_1 parameter selection. The λ_2 parameter is constant along the yellow lines. The optimal pair of λ_1 and λ_2 parameters is denoted by a green star. The colour represents the relative image error in this point: $\frac{\|x-x^*\|_2^2}{\|x^*\|_2^2}$, where x^* is the ground truth, x is the reconstructed image.

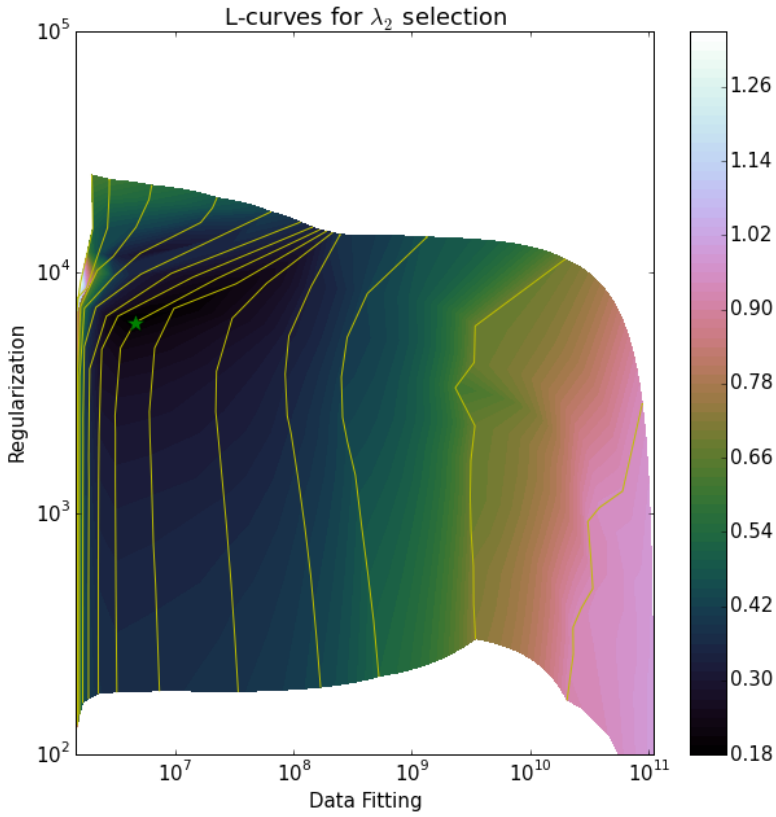

Figure 8.4: The modified L-curves plot for the Binary phantom for λ_2 parameter selection. The λ_1 parameter is constant along the yellow lines. The optimal pair of λ_1 and λ_2 parameters is denoted by a big green star. The colour represents the relative image error in this point: $\frac{\|x-x^*\|_2^2}{\|x^*\|_2^2}$, where x^* is the ground truth, x is the reconstructed image.

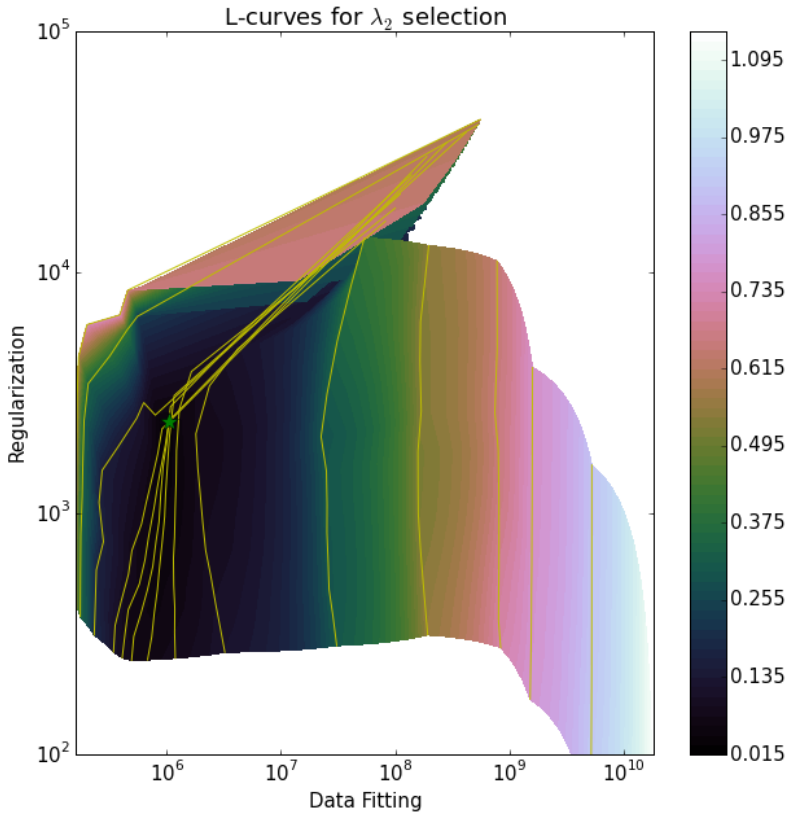

Figure 8.5: The modified L-curves plot for the Binary phantom for λ_2 parameter selection. The λ_1 parameter is constant along the yellow lines. The optimal pair of λ_1 and λ_2 parameters is denoted by a big green star. The colour represents the relative image error in this point: $\frac{\|x-x^*\|_2^2}{\|x^*\|_2^2}$, where x^* is the ground truth, x is the reconstructed image.

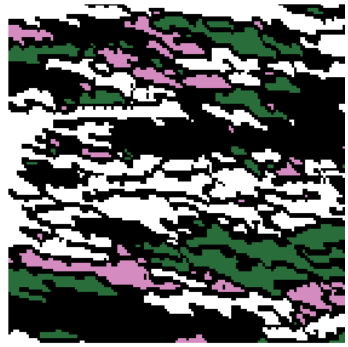

Figure 8.6: 4 class phantom for testing the L-curve approach testing.

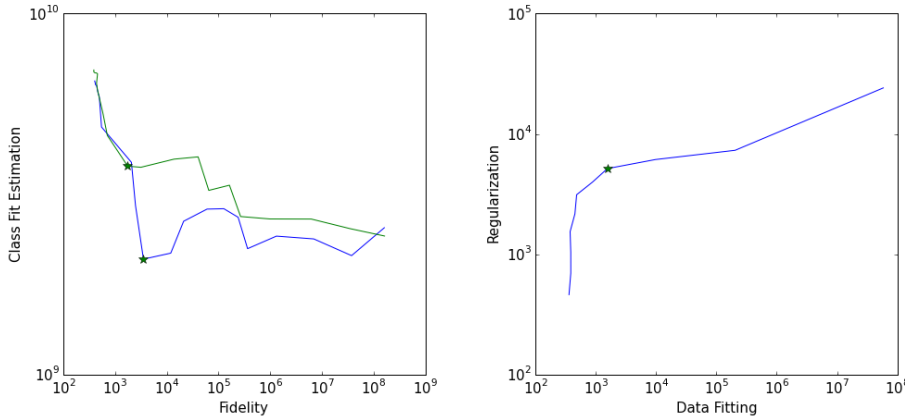

Figure 8.7: Testing of the regularization parameter selection approach. Left - curves for λ_1 parameter selection, blue - curve that corresponds to $\lambda_2 = 0.6$, green - curve that corresponds to $\lambda_1 = 0.78$, right - curve for λ_2 parameter selection. The green stars show the selected regularization parameters.

computed: $\frac{\|x-x^*\|_2^2}{\|x^*\|_2^2}$, where x^* is the ground truth image. The L-curves described in Section 8.3 are plotted

- in Figure 8.2 for the binary phantom for data fitting term and approximation of the class fitting term,
- in Figure 8.3 for the Shepp-Logan phantom for data fitting term and approximation of the class fitting term,
- in Figure 8.4 for the binary phantom for data fitting term and HMMFM regularization term,
- in Figure 8.5 for the Shepp-Logan phantom for data fitting term and HMMFM regularization term.

Consider the Figures 8.2 and 8.3. On these figures the yellow lines denote the lines, along which we do not change the λ_2 parameter changing only λ_1 regularization. Although these curves may look different, they have the shape that resembles the shape of the L-curve. Some of these lines have a corner. Also, we may note that the error value in the vicinity of the points where the L-curves change the behaviour is the smallest in all the figure (the error is denoted by a color). Thus, we may conclude that the regularization parameters that correspond to points in the vicinity of the corner of the L-curve produce good reconstruction. Thus, using this modified L-curves approach it is possible to select the λ_1 regularization parameter.

Next, consider the Figures 8.4 and 8.5. On these figures we do not change the λ_1 regularization parameter, changing only λ_2. We can see that the behaviour of these lines is different. We can see that the curves go from the left bottom corner up to the right top corner. In the beginning the curves go more or less along the axis. At some point the curve changes behaviour. We can see that the area where this happens corresponds to the optimal λ_2 regularization parameter. Thus, we have to find the λ_2 regularization parameter that corresponds to the point where the L-curve changes behaviour.

Using these approaches it is possible to find good regularization parameters λ_1 and λ_2 for the problem (8.6).

We have tested this approach on a test problem that contains 4 classes. The parameters of this problem are the same as the parameters of the problems considered above: the amount of the rays in the projection, the amount of projections,

the amount of noise in the data, the geometry of projections are exactly the same. The attenuation coefficients that are present in the image are 0.0, 0.33, 0.66, 1.0. The phantom may be found on Figure 8.6

Next, consider the graphs in Figure 8.7. We have started with finding the λ_1 regularization parameter with $\lambda_2 = 0.6$. The λ_1 was selected to be equal to 1.83. After that, having the λ_1 regularization fixed, the λ_2 regularization was selected according to the rule that was stated above. The regularization parameter was selected to $\lambda_2 = 0.78$. Next, the search for the regularization parameter λ_1 was run again. The selected regularization parameter was again equal to 1.83. Thus, we have found the pair of regularization parameters: $\lambda_1 = 1.83, \lambda_2 = 0.78$. The relative image error that was achieved with these parameters is equal to 0.16.

8.5 Conclusion

We have shown that an L-curve approach to the problem of selection good regularization parameters with some modifications gives some ideas about how to select good regularization parameters for the SRS algorithm. Although the problem of finding good regularization parameters for our algorithm is very complicated due to non-convexity of the problem (8.6), the minimal modifications based on intuitive understanding is a good heuristic in this case. We have validated this approach with two artificial problems. Although to select the second regularization parameter λ_2 we cannot use the L-curve criterion itself, there is a convenient substitution for this criterion that we have formulated above.

Bibliography

[1] Thomas Bonesky. "Morozov's discrepancy principle and Tikhonov-type functionals". In: *Inverse Problems* 25.1 (2009), p. 015015.

[2] Gene H Golub, Michael Heath, and Grace Wahba. "Generalized cross-validation as a method for choosing a good ridge parameter". In: *Technometrics* 21.2 (1979), pp. 215–223.

[3] Per Christian Hansen. "Analysis of discrete ill-posed problems by means of the L-curve". In: *SIAM review* 34.4 (1992), pp. 561–580.

[4] Per Christian Hansen. *Discrete inverse problems: insight and algorithms*. Vol. 7. Siam, 2010.

[5] Per Christian Hansen. *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion*. Vol. 4. Siam, 1998.

[6] Per Christian Hansen. *The L-curve and its use in the numerical treatment of inverse problems*. IMM, Department of Mathematical Modelling, Technical Universityof Denmark, 1999.

[7] Per Christian Hansen. "Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank". In: *SIAM Journal on Scientific and Statistical Computing* 11.3 (1990), pp. 503–518.

[8] Per Christian Hansen and Dianne Prost O'Leary. "The use of the L-curve in the regularization of discrete ill-posed problems". In: *SIAM Journal on Scientific Computing* 14.6 (1993), pp. 1487–1503.

[9] Per Christian Hansen and Maria Saxild-Hansen. "AIR tools—a MATLAB package of algebraic iterative reconstruction methods". In: *Journal of Computational and Applied Mathematics* 236.8 (2012), pp. 2167–2178.

[10] Maarten Jansen, Maurits Malfait, and Adhemar Bultheel. "Generalized cross validation for wavelet thresholding". In: *Signal processing* 56.1 (1997), pp. 33–44.

[11] Qinian Jin and Ulrich Tautenhahn. "On the discrepancy principle for some Newton type methods for solving nonlinear inverse problems". In: *Numerische Mathematik* 111.4 (2009), pp. 509–558.

[12] Puyan Mojabi and Joe LoVetri. "Adapting the normalized cumulative periodogram parameter-choice method to the Tikhonov regularization of 2-D/TM electromagnetic inverse scattering using Born iterative method". In: *Progress In Electromagnetics Research M* 1 (2008), pp. 111–138.

[13] Nhat Nguyen, Peyman Milanfar, and Gene Golub. "Efficient generalized cross-validation with applications to parametric image restoration and resolution enhancement". In: *Image Processing, IEEE Transactions on* 10.9 (2001), pp. 1299–1308.

[14] Mikhail Romanov et al. "Simultaneous Tomographic Reconstruction and Segmentation with Class Priors". In: *submitted to Inverse Problems in Science and Engineering* (2015).

[15] Dominique Van de Sompel and Michael Brady. "Simultaneous reconstruction and segmentation algorithm for positron emission tomography and transmission tomography." In: *ISBI*. 2008, pp. 1035–1038.

[16] Andrei Nikolajevits Tihonov. "Solution of incorrectly formulated problems and the regularization method". In: *Soviet Math* 4 (1963), pp. 1035–1038.

[17] Gennadii M Vainikko. "The discrepancy principle for a class of regularization methods". In: *USSR computational mathematics and mathematical physics* 22.3 (1982), pp. 1–19.

Chapter 9

Conclusion and Future Work

We consider the problem of computing a tomographic reconstruction and a corresponding segmentation in a joint operation based on the philosophy that the joint process will improve both the reconstruction and the segmentation. We developed a new algorithm called Simultaneous Reconstruction and Segmentation (SRS) in which the segmentation is based on given class priors. This information, in the form of means values and standard deviations for the classes, acts as prior for the reconstruction in the form of a Hidden Markov Measure Field Model (HMMFM).

We compared our approach to other techniques, and we find that the results obtained with our SRS algorithm (with given class parameters) produces better reconstruction and segmentation. We also tried two different regularization strategies spatial correlation of the classes: Tikhonov regularization and Total Variation regularization. We found that the results of these two strategies are equally good, but the Tikhonov regularization strategy leads to a simpler and faster implementation.

We introduced some modification to this approach such as relaxation of the regularization parameter (λ_1 decrease with iterations) and relaxation of the standard deviations (σ_k decrease with iterations) of the classes. We also compared the

effect of using different values for the standard deviations and mean values of the classes. Moreover we found a good heuristics for finding good regularization parameters for the SRS algorithm without knowing the ground truth.

Still, there are a lot of things to look at. This includes: **Use of another class regularization term.** We have used a Gaussian class-fitting term, since it is a common way of approximating an unknown distribution. This term has the form:

$$\log p(\mathbf{x}|\delta) = \sum_j \log \sum_k \delta_{jk} \exp\left(-\frac{(x_j - \mu_k)^2}{2\sigma_k^2}\right). \tag{9.1}$$

It would be also interesting to look at another class-fitting term of the form:

$$\log p(\mathbf{x}|\delta) = \sum_j \log \sum_k \delta_{jk} \exp\left(-\frac{|x_j - \mu_k|^2}{\sigma_k}\right) \tag{9.2}$$

which will lead to a LASSO-type regularization of the pixel values. **Spatial regularization of the pixels in the reconstruction.** In the present work the spatial regularization is applied to the classes only, in order to see how well this regularization was able to influence the reconstruction. Another direction of research is therefore to incorporate in the SRS algorithm some form of spatial regularization of the pixels in the reconstruction. It is expected that spatial regularization of the reconstruction may (significantly) improve the reconstruction (cf. Tikhonov regularization and Total Variation regularization). Combining our approach with spatial regularization of the reconstruction may reduce the amount of artifacts that appear in the solution when only the spatial regularization is present. As a spatial regularization the following terms may be considered:

$$\mathrm{Reg}(x) = \sum_j \|D_j \mathbf{x}\|_2 \tag{9.3}$$

$$\mathrm{Reg}(x) = \sum_j \sum_{j' \in N(j)} \left| \sum_k \delta_{jk} \frac{x_j - \mu_k}{\sigma_k} - \sum_k \delta_{j'k} \frac{x_{j'} - \mu_k}{\sigma_k} \right|. \tag{9.4}$$

More tests. It is also interesting to test this approach on more different real data from CT scanners.